Tivoli® Application Dependency Discovery Manager/Tivoli Business Service Manager Integration Solution

Including a Virtual Center and a Monitored Custom Application Example

Guide Version 2.0

Phil Riedel
Tivoli Service Availability and Performance Management Integration Center of Competency

© Copyright International Business Machines Corporation 2011 and 2012.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

CONTENTS

List of figures v

1 Introduction 8

2 Who should read this guide 8

3 Assumptions 8

4 What this guide includes 8

4.1 TADDM 7.2.1 streaming deployment 10

4.2 Discovery Library Toolkit (if003) 12

5 Configuring the DLT for a JDBC connection 13

Transfer the JAR files 13

Add the change table to the TADDM database 13

Final Steps to pull data from TADDM 13

6 Virtual center integration 15

6.1 TADDM discovery of virtual centers 15

7 Defining business services 19

7.1 Adding a TBSM created business service to TADDM 20

8 Creating a business service dashboard 21

8.1 TADDM Configuration 21

8.1.1 Creating a Custom Server Template (CST) 22

8.1.2 Running a discovery to collect your CST objects 24

8.1.3 Grouping Composer 25

8.1.4 Grouping composer hierarchical example 28

8.1.5 Kickoff a Level 2 or Level 3 Discovery 31

8.2 TBSM Configuration 32

8.2.1 Editing the rules file for custom applications 32
8.2.2 Running the DLT 33
8.2.3 Dashboard pre-configuration 33
8.2.4 Add data source for PostgresDB 33
8.2.5 Adding a Data Fetcher 34
8.2.6 Data fetcher view results for staf custom server 35
8.2.7 Create a template to be used by the Dashboard Service Portlet 35
8.2.8 Create a service which uses this template 37

8.3 Portlet Creation Step by Step 41

8.3.1 Create a new page 42
8.3.2 Set portlet type to Service Tree 42
8.3.3 Save the page with name “staf-status” 44
8.3.4 Edit the defaults for your new Service Tree 45
8.3.5 Set starting instance to Business Application (STAF) 46
8.3.6 Save your changes between steps 46
8.3.7 Edit Page to add additional portlets 47
8.3.8 Perform a vertical split 47
8.3.9 Choose AEL Portlet 48
8.3.10 Split both portlets Horizontally 49
8.3.11 Save the new page again 50
8.3.12 Set context default preferences for lower right to the StafCST imported service 51
8.3.13 Select viewer type to “Basic Relationships” 52
8.3.14 Select lower left context to TestSub 52
8.3.15 Lower Left portlet viewer type set to “Relationships” 53
8.3.16 Remove all wires 54
8.3.17 Create one new nodeClickOn wire 54
8.3.18 The Dashboard (widget) portlet 55
8.3.19 Selecting your indicator/widget type 62

9 Changing the State GIF Images 63

10 Time Window Analyzer for STAF 66

11 Tivoli Monitoring configuration (for STAF) 69

11.1 Create a situation event for STAF 70

11.2 Create a new Situation 71

11.3 Add the process command to identity the STAF process 72

11.4 Assign the situation to the systems where STAF runs 73

11.5 Set situation severity and target probe 74

12 TADDM: Cleaning up dormant components 75

13 Summary 76

14 Appendix 76

14.1 Appendix A. Architecture 76

14.2 Appendix B. Information centers 77

14.3 Appendix C. Acronyms 77

15 Notices 78

15.1 Trademarks 80

LIST OF FIGURES

Figure 1: TADDM 7.2.1 Streaming configuration with TBSM 9
Figure 2 TADDM Servers Summary 11
Figure 3 The Component Registry Viewer tool (crviewer) 12
Figure 4 TADDM Virtual Center Topology 16
Figure 5 TBSM Virtual Center Service Viewer 16
Figure 6 TADDM Access List Entry 17
Figure 7 TADDM Virtual Center Topology (Enlarged) 17
Figure 8 TBSM Service Component Repository View 18
Figure 9 Business Application Best Practice 19
Figure 10 Sample dashboard 21
Figure 11 Define Custom Server 22
Figure 12 Define the Configuration File for TADDM to discover 23
Figure 13 Starting a discovery in TADDM 24
Figure 14 Navigate to the Grouping Composer 25
Figure 15 Create a “New” group. 25
Figure 16 Grouping Composer General Information 26
Figure 17 Grouping Composer Rules for STAF 27
Figure 18 StafNetwork rule-based content 28
Figure 19 Grouping Composer for Hierarchy 29
Figure 20 Creating a Business Service containing Business Applications 30
Figure 21 TBSM Display of Business Service Hierarchy 30
Figure 22 More Complex Naming Example 30
Figure 23 Hierarchy Topology in TADDM 31
Figure 24 Adding Custom Application support for DLT 32
Figure 25 Data Source Creation 33
Figure 26 Data Fetcher Query for Application 34
Figure 27 View Query Results 35
Figure 28 Create a Template 35
Figure 29 Incoming status rule selector 36
Figure 30 Incoming status rule 37
Figure 31 TestSub uses the STAFRule Template 38
Figure 32 Set scrid filter for incoming status rule 39
Figure 33 Create Parent object for the dashboard portlet service 40

Figure 34 TBSM tree view of the dashboard service 40
Figure 35 Widget portion of the dashboard 41
Figure 36 Create a new custom page 42
Figure 37 Select the Service Tree Portlet 42
Figure 38 View of new portlet 43
Figure 39 Saving the new portlet page 44
Figure 40 Defaults Edit 45
Figure 41 Set STAF as the starting instance 46
Figure 42 Save the changes – found in the lower left corner 46
Figure 43 Re-edit the page to create additional portlets 47
Figure 44 Vertical split 47
Figure 45 Choose a portlet type 48
Figure 46 Assign portlet types to the new portlets 49
Figure 47 Service Viewer for StafCST 51
Figure 48 Basic Relationships Viewer 52
Figure 49 Dashboard service portlet settings 52
Figure 50 Service Viewer Settings 53
Figure 51 Remove “default” wires 54
Figure 52 Add a new wire 54
Figure 53 Create a wire using drag and drop 55
Figure 54 Current dasbord portlet view 55
Figure 55 Defining a new service viewer using indicators 56
Figure 56 Add an indicator dashboard element 57
Figure 57 Indicator menu service instance 57
Figure 58 Enter the visual representation for this indicator 58
Figure 59 The indicator now contains count of STAF servers and the range (20) 58
Figure 60 Set displayName to “StafServers” 60
Figure 61 Complete widget dashboard 61
Figure 62 Widgets which support bi-directional negative status 62
Figure 63 Service Tree State GIFs 63
Figure 64 Tree Template Editor 63
Figure 65 Select Template name and Edit Policy 64
Figure 66 Edit Policy for gif’s 65
Figure 67 Time Window Analyzer 66
Figure 68 Record Template Name and Rule Name 67
Figure 69 Time Window Analyzer for STAF services count (ct) 68
Figure 70 Configuring Tivoli Monitoring to send Events 69

Figure 71 From the enterprise portal add a situation 70
Figure 72 Set the situation name to “StaffMissingSituation” 71
Figure 73 Set Process command as shown 72
Figure 74 Assign to ALL_UNIX 73
Figure 75 Set the EIF alert receiver 74
Figure 76 Dormant Component Filter 75

1 Introduction

IBM® Tivoli® Application Dependency Discovery Manager (TADDM) and Tivoli Business Service Manager (TBSM) are designed to work together, and their primary functions complement each other. Configuring this integration requires some steps to be taken during and after installation. This solution guide provides an overview of those steps, and their benefits.

This solution guide demonstrates the Virtual Center support and shows the ability to update TBSM portlets for custom servers and business applications that are discovered or created by TADDM. The Virtual Center sensor discovers VMware details for the following versions:

- VMware Virtual Center Server Version 2.5
- VMware vCenter Server Version 4.0 and 4.1

This guide demonstrates TADDM Version 7.2.1 and TBSM Version 4.2.1 with TBSM discovery library toolkit (DLT) IF003. The toolkit is configured with a JDBC connection, which is the fastest available interface method with TADDM. The toolkit is used to load data into TBSM from discovery library adapter (DLA) files or by pulling configuration management data base (CMDB) data from TADDM.

2 Who should read this guide

If you are using TADDM, TBSM, or both products, this solution guide provides valuable information and solution examples, use cases and configuration examples.

3 Assumptions

This guide assumes you have installed TADDM, TBSM and the DLT at the levels listed in the introduction. This solution guide assumes a DB2 database backend is in use with TADDM. See the product documents for examples if using Oracle.

4 What this guide includes

This guide includes the following information:

- Introduction of TADDM Version 7.2.1 and TBSM DLT Version IF003
- Introduction to and support for virtual centers
- An example of a monitored business service

TADDM Steps

- Custom Server Template
- Custom Query

- Grouping Composer
- Managing Dormant Components

TBSM Steps

- Rules file changes for Custom Server Templates
- Data Source, Data Fetcher, Template and Service
- Portlet creation step by step

Other Steps

- IBM Tivoli Monitoring Integration for STAF application

Since the release of version 1.0 of this guide, there have been improvements in the integration of TBSM DLT and TADDM. This is a brief overview of these improvements as it applies to these solutions examples and the suggested configurations applicable to this solution guide.

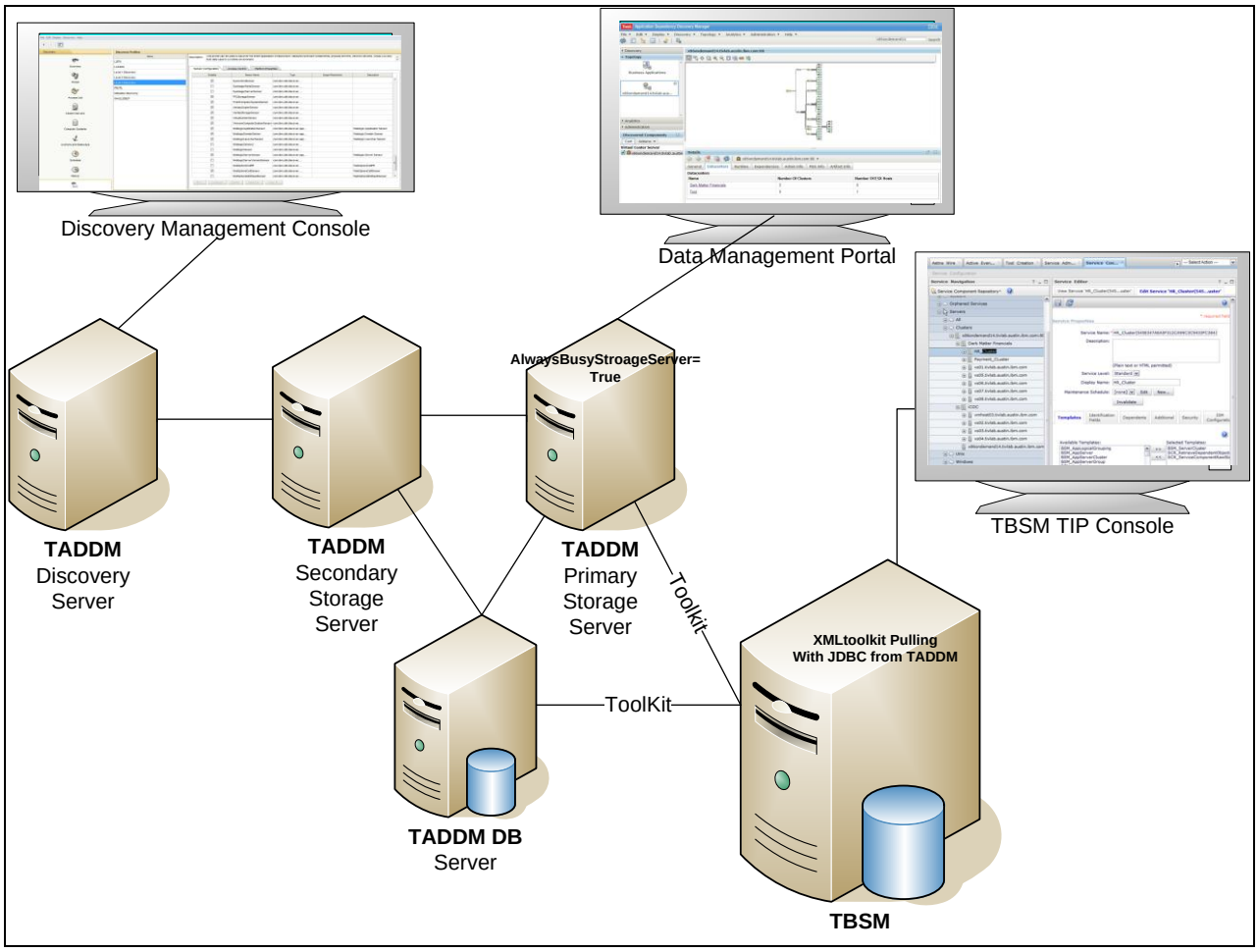


Figure 1: TADDM 7.2.1 Streaming configuration with TBSM

4.1 TADDM 7.2.1 streaming deployment

The TADDM streaming environment offers several key advantages over previous single system or enterprise sync-mode TADDM configurations. Using a TADDM streaming 7.2.1 environment (shown in Figure 1: TADDM 7.2.1 Streaming configuration with TBSM) workload is distributed while operating from a single database. TADDM 7.2.1 single system (not shown) and streaming environments can run concurrent discoveries and IDML book loads. The discovery management console is streamlined to provide only discovery related tasks. And the data management portal in this configuration is dedicated to display the configuration management data that is collected by TADDM discovery and loaded by the IDML books. The primary storage server is dedicated for display by the `AlwaysBusyStorageServer=true` property in the %TADDM_HOME%/etc/collation.properties file. The image below shows how this property setting removes a data management portal from the group of storage pool members.

With TADDM 7.2.1 you no longer have to use application templates or application descriptors to define business applications. Using the data management portal, you can now use the grouping composer by clicking **Discovery > Grouping Composer** to define collections, access collections and business applications. If running a standalone TADDM environment you can use the Grouping Composer to define business services, however in an environment with TBSM, use TBSM for business service creation.

TADDM 7.2.1 also introduces improvements to its state event capabilities with new sendOriginGuid and sendCauses attributes.

In this solution guide TBSM is assumed to be connected to an event object server, and both TADDM and IBM Tivoli Monitoring are also configured to send events to the EIF probe that is associated with this object server.

If you have Tivoli Monitoring agents deployed across your enterprise you should consider performing your discovery through Tivoli Monitoring. Tivoli Monitoring discovery simplifies the need to manage your access control list.

For more information on this type of discovery see *TADDM Discovery using the IBM Tivoli Monitoring Scope Sensor*:
https://www.ibm.com/developerworks/wikis/display/tivoliaddm/TADDM+Discovery+using+the+IBM+Tivoli+Monitoring+Scope+Sensor

For more general information on TADDM 7.2.1 see the TADDM Information Center:
http://publib.boulder.ibm.com/infocenter/tivihelp/v46r1/index.jsp?topic=/com.ibm.taddm.doc_721/welcome_page/welcome.html

Figure 2 TADDM Servers Summary

The TADDM Servers Summary page can be seen by clicking Analytics then clicking TADDM Servers Summary. This page shows the state of your TADDM environment and whether your Primary Storage Server is configured as a display server or as a display and storage pool member.

4.2 Discovery Library Toolkit (if003)

With interim fix IF003, the TBSM toolkit can now load data directly from the TADDM database using a JDBC connection. This method is approximately 66% faster than the previous DLT load using APIs. As a result, it is even easier to maintain the “real-time” status of your configuration items in TBSM. In addition, TBSM added several filter changes and model changes to improve the ease of loading key TADDM configuration items.

This version of the toolkit also provides a powerful new tool for viewing the objects you loaded into TBSM through the pull from TADDM or through discovery library adapter files into TBSM. The Component Registry Viewer tool (crviewer) is not covered here. The following figure contains an example. You can view the following short video for more information.

http://www.youtube.com/watch?v=ZVCxcKZSOz0

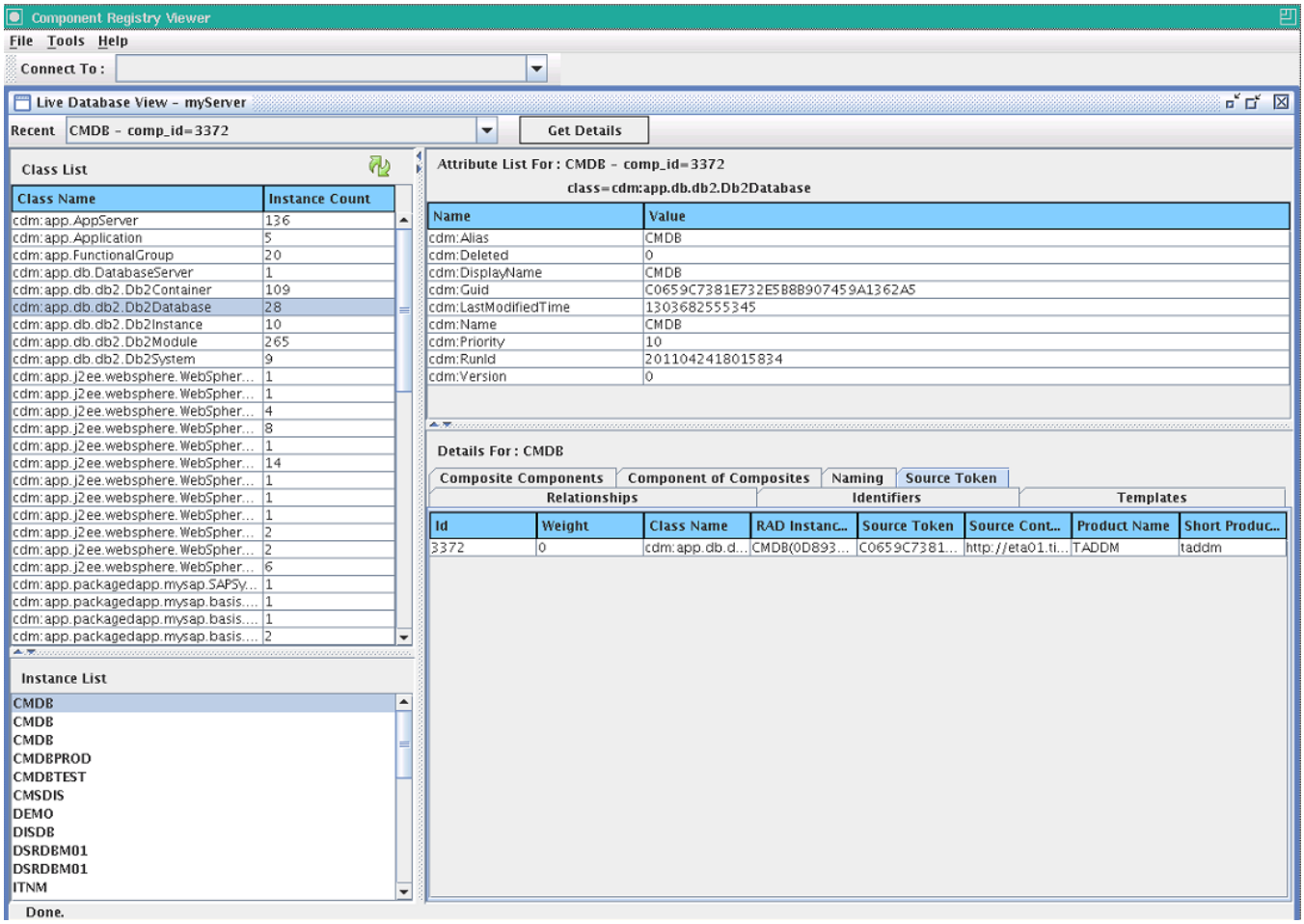


Figure 3 The Component Registry Viewer tool (crviewer)

5 Configuring the DLT for a JDBC connection

Do the following process to configure TBSM DLT for a JDBC connection with TADDM.

Transfer the JAR files

The TADDM JAR files that are required by the toolkit are obtained by the DLT automatically. Except a new JAR required for the JDBC connection (marked with an * below). Follow these instructions if you would like to obtain all of these manually, or just obtain oal-topomgr.jar manually:

1. From TADDM, copy the following JAR files from the %TADDM_HOME%/dist/sdk/lib directory:
   oal-topomgr.jar*
   platform-model.jar
   taddm-api-client.jar

2. Place these files into the %TBSM_HOME%/XMLtoolkit/sdk/clientlib directory.

If you start the toolkit and one or more new TADDM jars are detected, the jar will be pulled automatically and you will be asked to stop and restart the toolkit to continue operations. Please see the XMLtoolkit/logs/msgGTM_XT.log.0 file for these messages.

Add the change table to the TADDM database

A JDBC connection requires a change table be added to the TADDM database server for TBSM.

1. From the %TBSM_HOME%/XMLtoolkit/sql directory copy the `taddm_schema_setup_db2.sql` file to the TADDM database server.

2. Using your DB2 instance user, connect to the TADDM database and enter `db2 -tvf taddm_schema_setup_db2.sql`.

If using an Oracle database for TADDM please use the oracle file in a similar fashion.

Final Steps to pull data from TADDM

1. As in the past, you might have to run the **`setxmlaccess`** script to update your user information. This command can be found in the %TBSM_HOME%/XMLtoolkit/bin directory.

2. For JDBC, you must issue the following command to set up the direct connection (found in the same directory as the **`setxmlaccess`** script (only required if you did not configure TADDM at the time of install or your configuration has changed):

```
./taddmdirect.sh -enable -t DB2 -h obiwan.xyz.raleigh.ibm.com
-d taddm -p 50000 -s db2inst1

Parameter descriptions:
-t database type DB2 or ORACLE
-h fully qualified host name of the database
-d database name
-p db port
-s instance user

For more information see the TBSM TLK documentation

http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?topic=%2Fcom.ibm.tivoli.itbsm.doc%2Fwelcome.htm search for
taddmdirect.sh
```

This call assumes a DB2 database for TADDM, and db2inst1 as the TADDM database user.

3. Start the DLT from the same directory using **`tbsmrdr_start`** command.

Note: If TADDM is selected during toolkit install, steps 2 and 3 will be performed for you.

6 Virtual center integration

6.1 TADDM discovery of virtual centers

If you want to use TADDM to discover virtual centers (VC), this section guides you with any necessary steps. You can also see how the VC systems are represented in both TADDM and TBSM.

With TADDM 7.2.1, you perform the discovery steps from a discovery console and the viewing steps from the data management portal.

To locate the virtual center on the x86ondemand14 host, do the following steps:

1. On the TADDM Data Management Portal, enter the host name into the **Search** field and click **Enter.**

2. Select the Virtual Center filter from the results page.

3. Select the component and click **Add to cart**. Then from the discovered components Cart, select the check box next to this entry and click Show Virtual Systems Topology.

Figure 4 TADDM Virtual Center Topology shows the TADDM Data Management Portal with the topology for the Virtual Center Server displayed. Figure 5 TBSM Virtual Center Service Viewer shows a similar view of the same object in TBSM. This TBSM view is available after performing the discovery steps in TADDM and after the next DLT run.

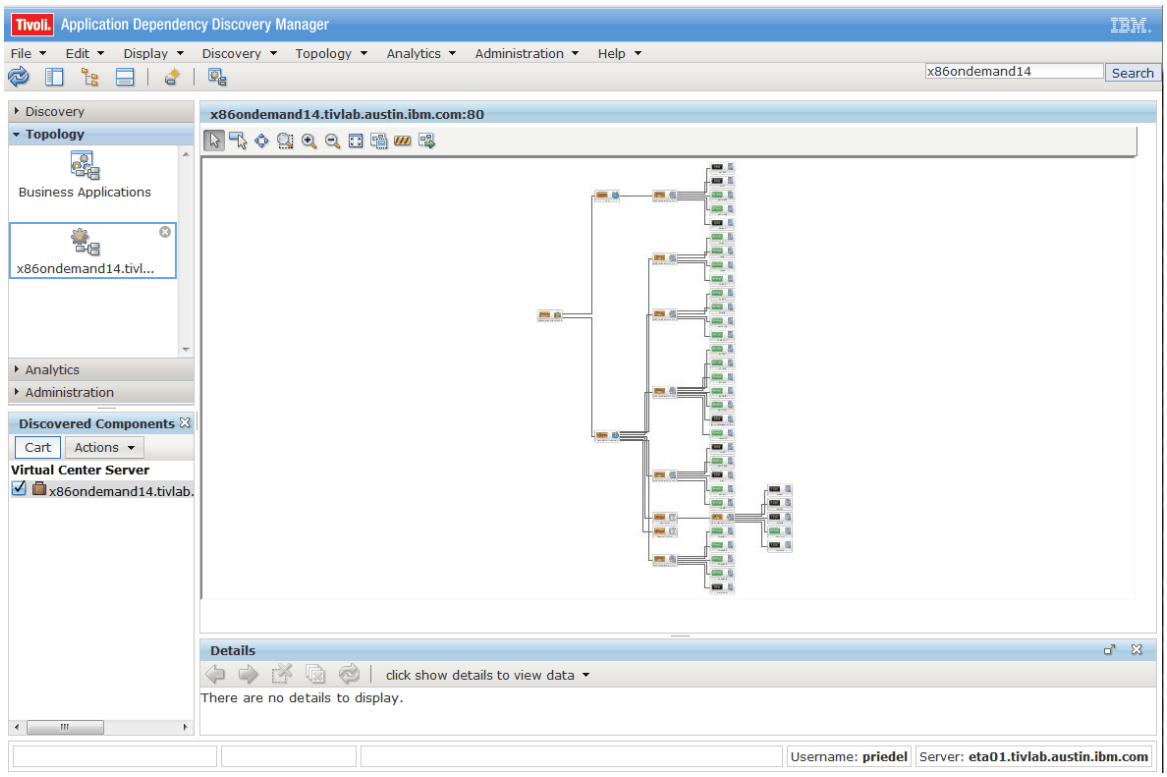


Figure 4 TADDM Virtual Center Topology

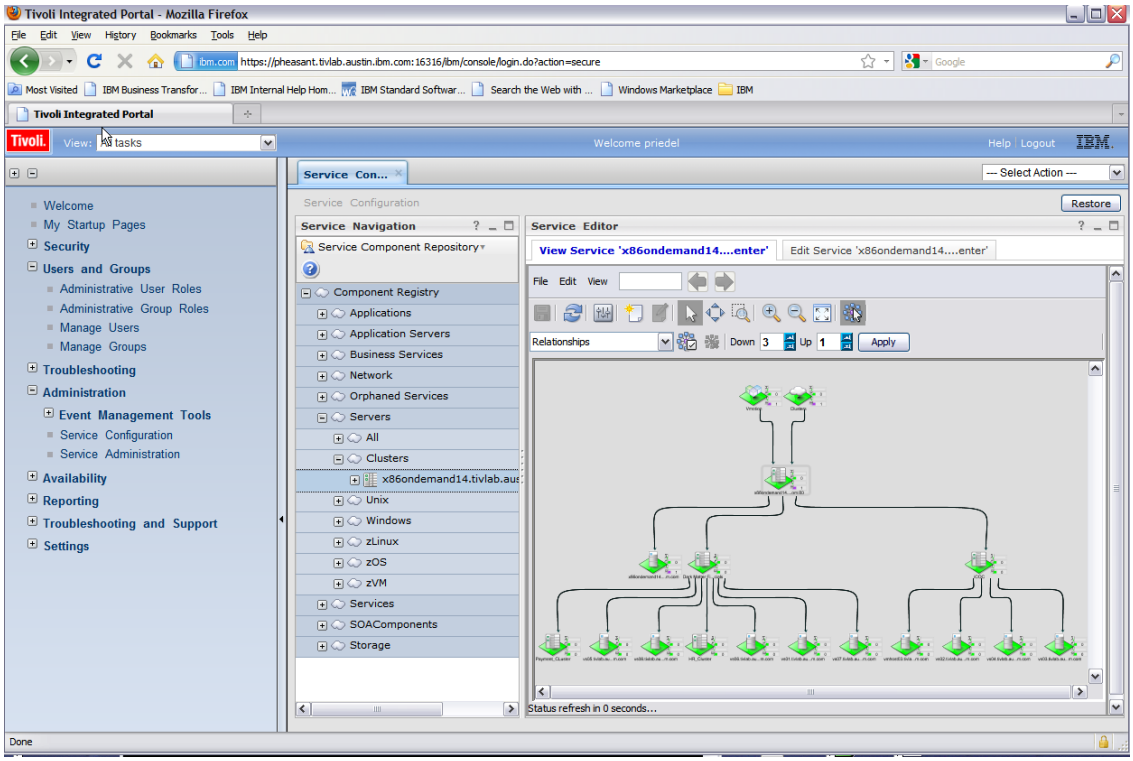


Figure 5 TBSM Virtual Center Service Viewer

In version 1.0 of this guide you had to perform steps to enable the virtual center sensor. In TADDM 7.2.1 you only have to run a level 2 or level 3 discovery. Prior to running a discovery, add an access list entry for the virtual center as shown in Figure 5. Also add a scope including the virtual center host (not shown).

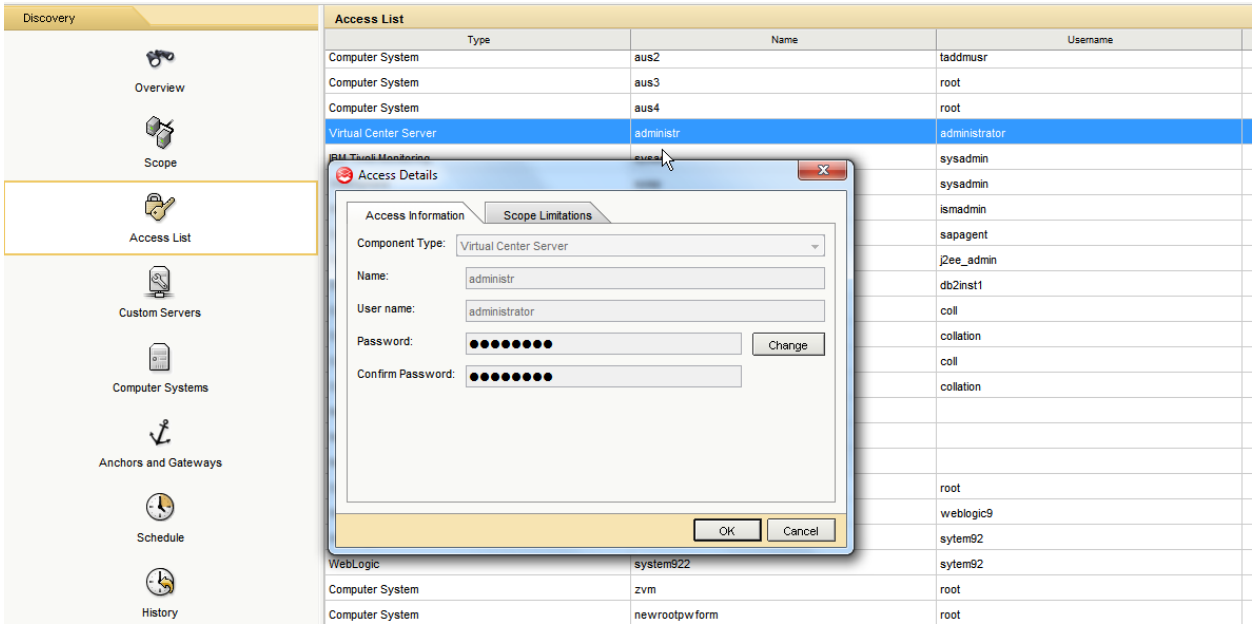


Figure 6 TADDM Access List Entry

Figure 6 is a TADDM Topology View of a virtual center (partial), showing virtual topology on the Virtual Center Server CI named x86ondemand14.tivlab.austin.ibm.com.

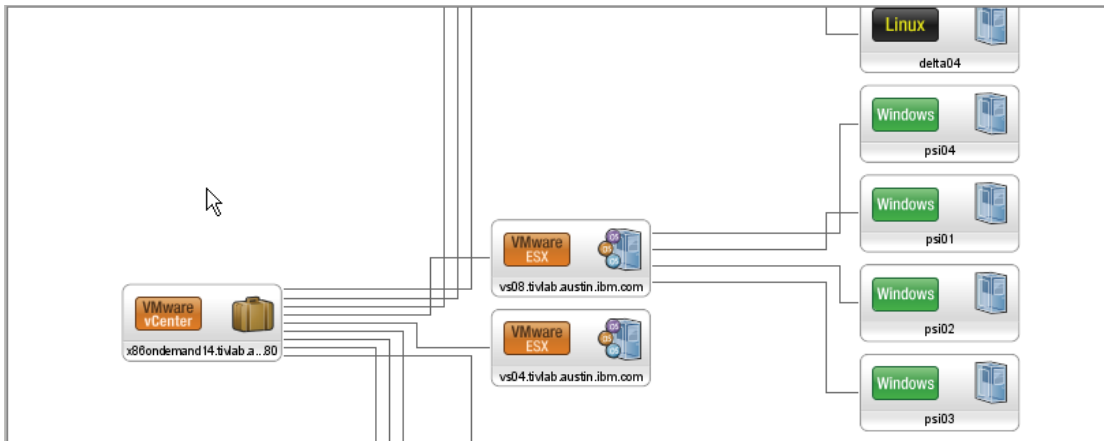


Figure 7 TADDM Virtual Center Topology (Enlarged)

With TBSM DLT IF003, you do not have to make any changes to pull Virtual Center CIs and relationships in TBSM. If you have migrated to IF003 it is necessary to move or merge the changes in the xml files in directory /XMLtoolkit/xml/install into /XMLtoolkit/xml. The support is now

included for new installations post IF003. Below is an SCR view for Virtual Center on the TBSM Tivoli Integrated Portal.

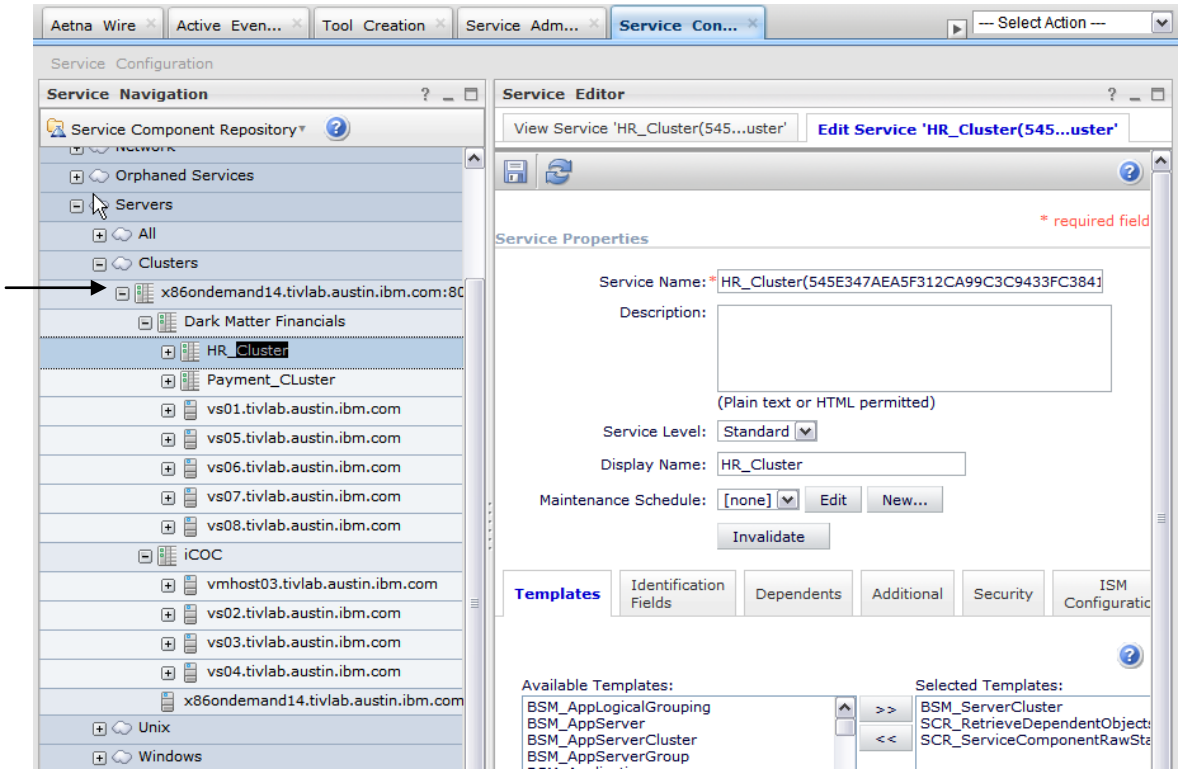


Figure 8 TBSM Service Component Repository View

With virtual environments, change can occur more rapidly than your TADDM discoveries may reflect. To capture these virtual-motion changes both in TADDM and TBSM you can take advantage of the pro-active update functions provided by TADDM. For more details on this solution enter this URL:

https://www.ibm.com/developerworks/mydeveloperworks/blogs/7d5ebce8-2dd8-449c-a58e-4676134e3eb8/entry/using_itm_to_send_proactive_updates_to_taddm82?lang=en

7 Defining business services

In version 1.0 of this guide the best practice was to use TBSM for defining business services. Using this method requires the TBSM DLA to load those "TBSM created" business services into TADDM. With the advent of the grouping composer our recommendation is to continue this practice using the grouping composer to create only Business Applications and Collections using TADDM.

Use the TADDM Grouping Composer

New for 7.2.1, use the Grouping Composer to define your Business Applications in TADDM

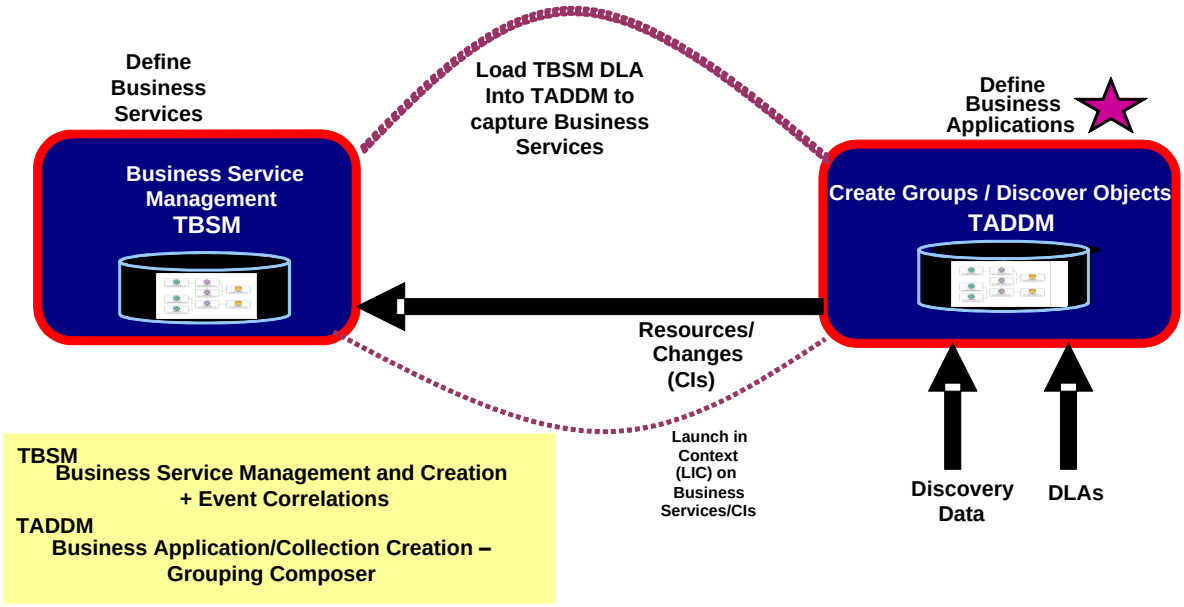

Define Business Services
Load TBSM DLA Into TADDM to capture Business Services
Define Business Applications
Business Service Management TBSM
Create Groups / Discover Objects TADDM
Resources/ Changes (CIs)
Launch in Context (LIC) on Business Services/CIs
Discovery Data
DLAs
TBSM
Business Service Management and Creation + Event Correlations
TADDM
Business Application/Collection Creation – Grouping Composer

Figure 9 Business Application Best Practice

After creating business services in TBSM, to represent these relationships in TADDM it will be necessary to run the TBSM discovery library adapter and then load that IDML into TADDM. It is now possible to define business services in TADDM, removing the need for the DLA, but TADDM does not contain the same richness of function that TBSM does for this purpose.

7.1 Adding a TBSM created business service to TADDM

Use the following procedure to add TBSM created business services to your configuration management data base.

1. On TBSM system change current directory to %tbsmhome%/XMLtoolkit/bin

2. Run the TBSM DLA command shown: **genidml.sh -U bsmuser -P bsmupw -o /tbsm/XMLtoolkit/exportdla**

3. Copy the output file to your TADDM server.

4. Start the TADDM non-root user. Change the directory to %taddmhome%/dist/bin.

5. Run the **./loadidml.sh –f <TBSM DLA file>** file name **itbsm..refresh.xml** command.

8 Creating a business service dashboard

This section shows the steps you do to create the dashboard shown in figure 9. This includes the Custom Server Templates, TADDM discovery, grouping composer configuration, TBSM rules file update, and the TBSM console steps.

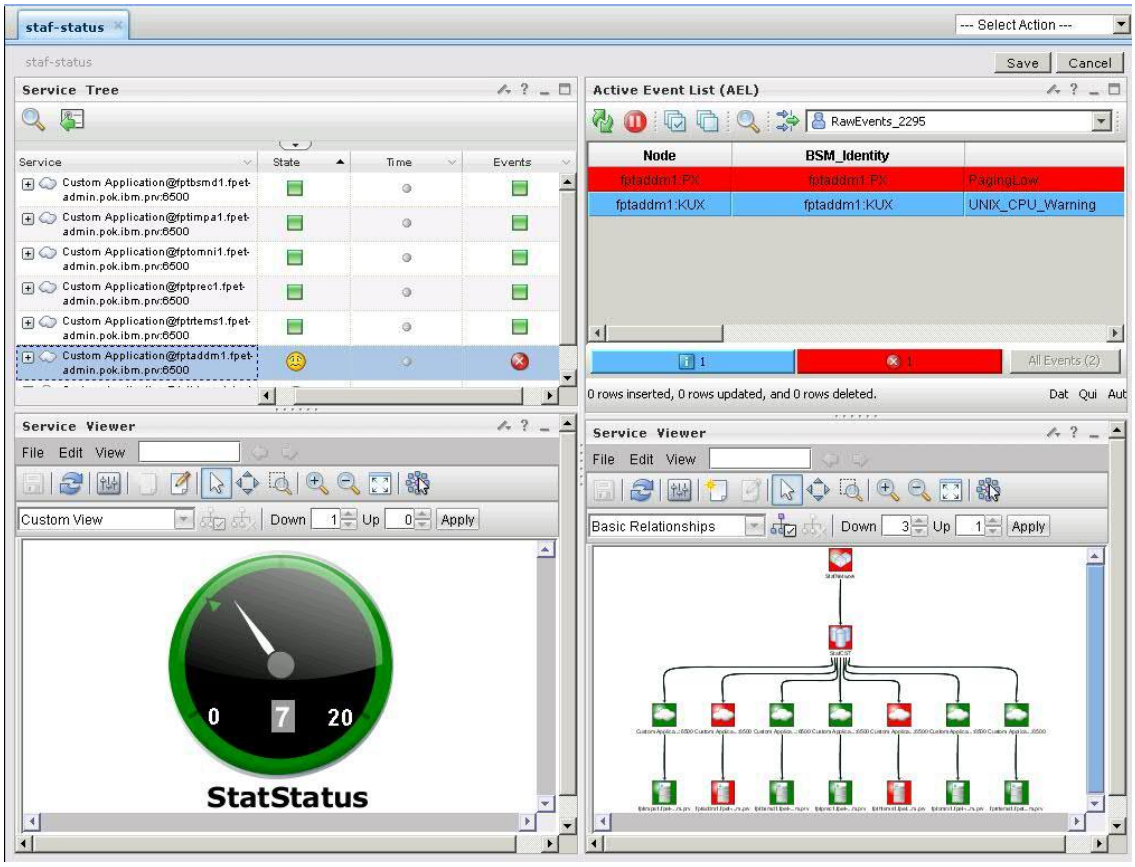


Figure 10 Sample dashboard

8.1 TADDM Configuration

This solution uses an internal application called STAF as the basis for these examples. These examples are equally applicable to Human Resources, Payroll, or other key applications in your enterprise.

8.1.1 Creating a Custom Server Template (CST)

A custom server template contains descriptive criteria that are used to assign unknown server processes to the custom server. You specify these criteria when defining the template for the custom server. Custom servers are any servers that do not already have sensors in TADDM, for example your payroll application, banking applications, or the example noted in this solution guide, an application or process called SysTestAutoFrameworkSTAF.

1. From the data management portal, click **Discovery** > **Custom Servers > New**.

2. Set the new custom server template as enabled.

3. Select the custom icon that you want to use.
4. Complete the information: Type is **AppServer**, Action is **Discover**. In this example the Identifying Criteria is set with **Program Name contains STAFProc**. STAFProc is the program name that is started when running the STAF application.

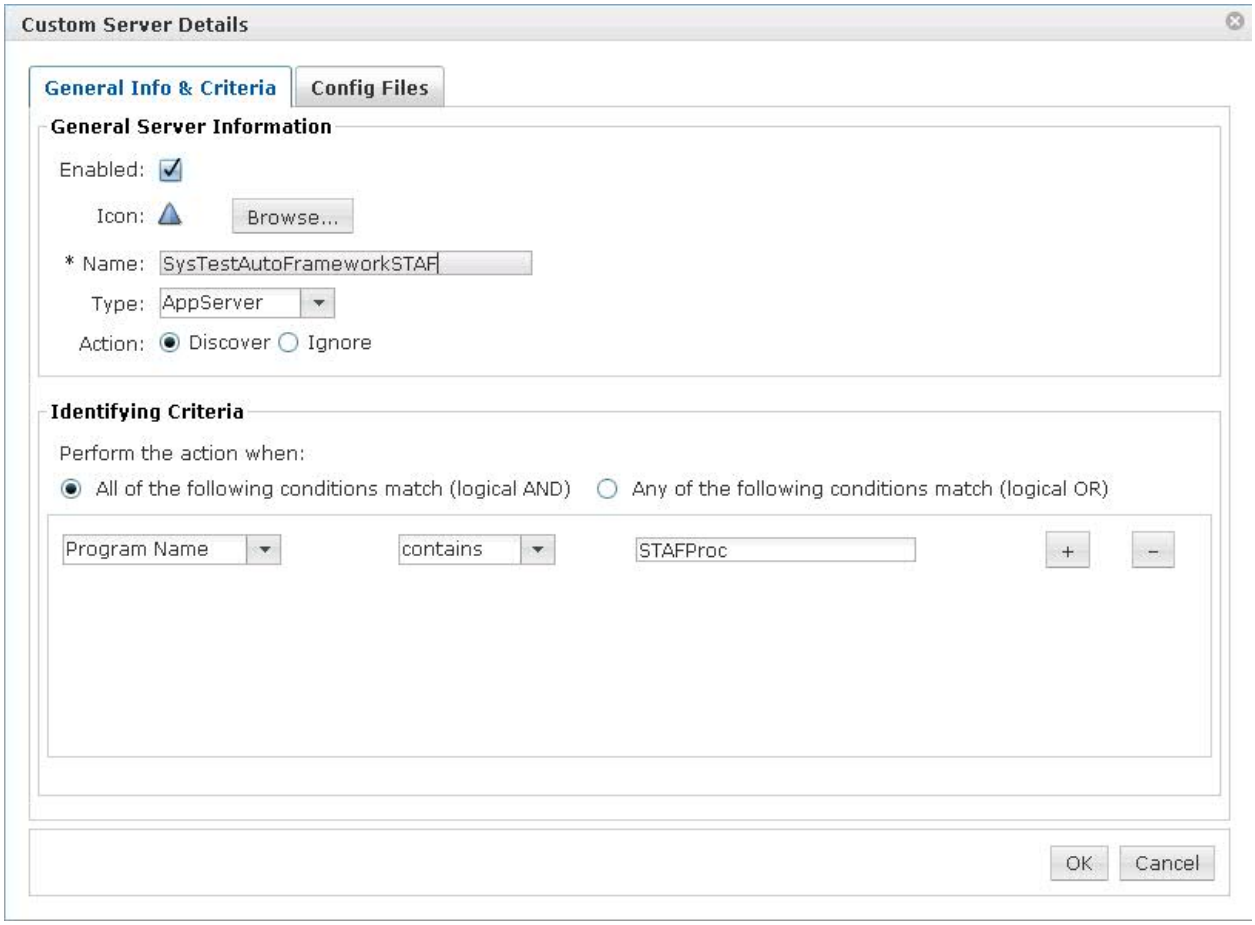


Figure 11 Define Custom Server

When defining custom server templates, you can also capture key configuration files that are used by the application. In this example you collect the STAF profile. Do the following steps to do this:

1. Click the **Config Files** tab.

2. Create an entry as shown in Figure 12 Define the Configuration File for TADDM to discover. Collecting these files allow the TADDM change history to reflect changes in these key files.

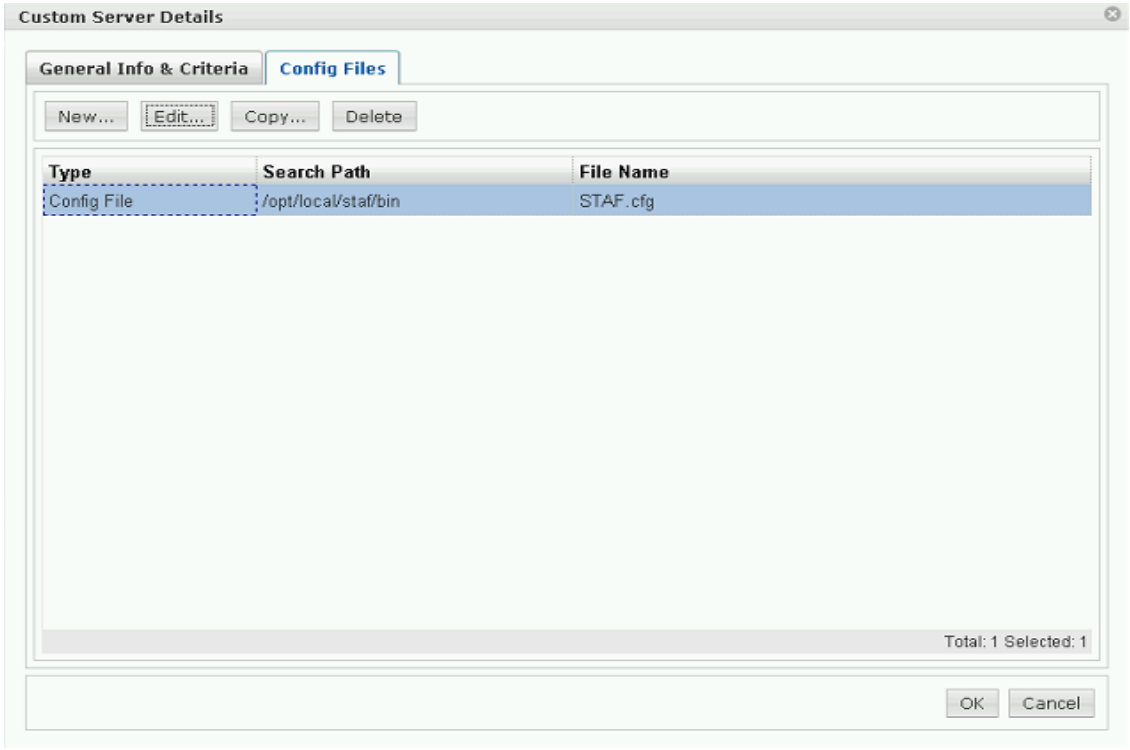


Figure 12 Define the Configuration File for TADDM to discover

In version 1.0 of this solution guide the next step was to create an application template. Application templates have been replaced by the grouping composer. In this version of the guide, the next step is to run a discovery to populate instances of the custom application that you just defined.

To run a discovery, do the following steps:

1. Log into the discovery management console.

2. From the discovery overview page, run a level 1 or level 2 discovery against the target systems that contain your application. If the STAF application its running on hosts selected in Figure 13 Starting a discovery in TADDM.

8.1.2 Running a discovery to collect your CST objects

Run a discovery selecting the hosts where your custom servers are located. Click Run Discovery then click OK.

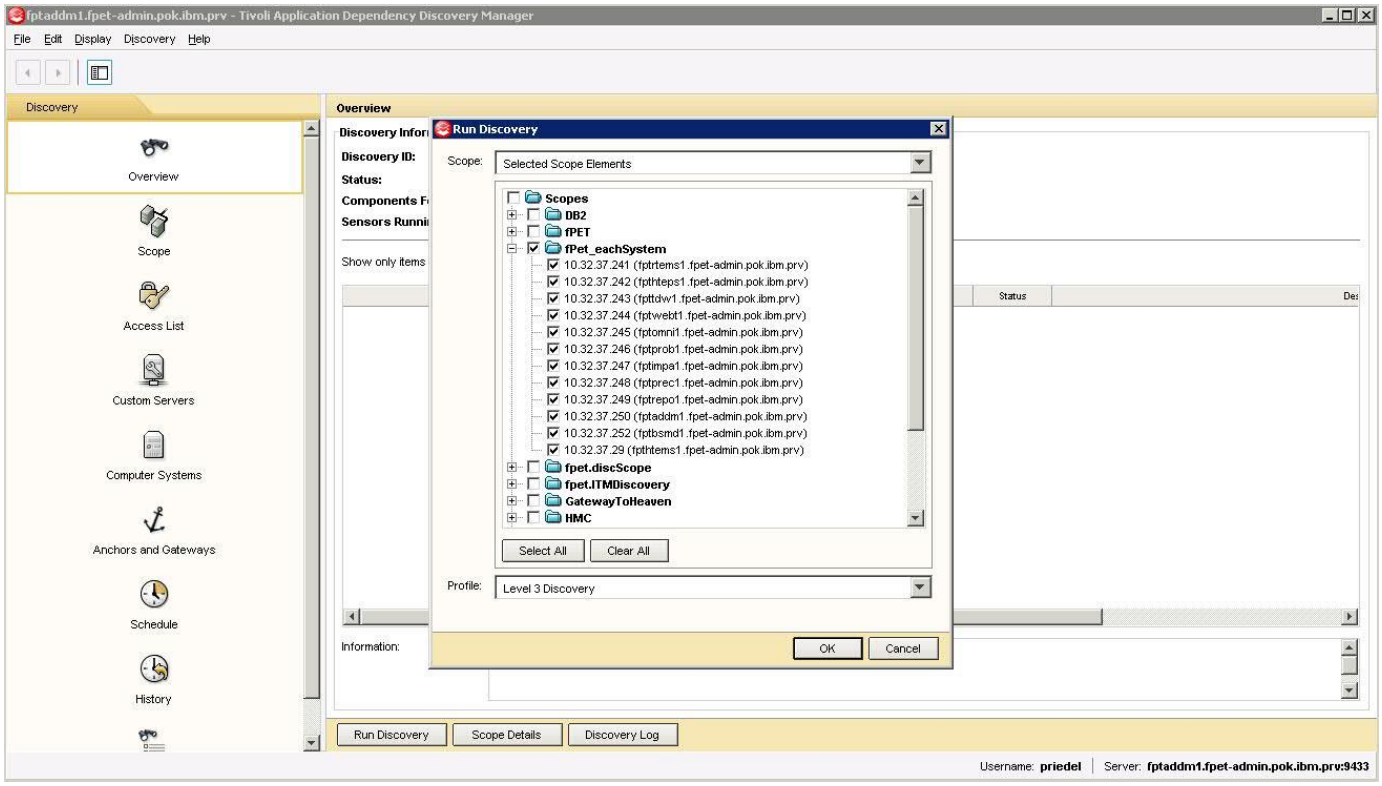


Figure 13 Starting a discovery in TADDM

8.1.3 Grouping Composer

You can now create a grouping composer object using your custom server template CIs.

1. On the Data Management Portal, click **Discovery > Grouping Composer.**

Figure 14 Navigate to the Grouping Composer

2. In the Data Management portal, Click **New**.

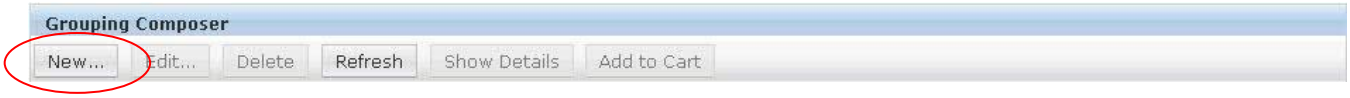


Figure 15 Create a "New" group.

3. In the Group Details window, click the **General Information** tab.

4. Complete the following information:

   Name: StafNetwork – a subset of the CST object name is used here, any name could be used
   Type: Business Application
   The Description and URL fields are optional.

5. Click **Next.**

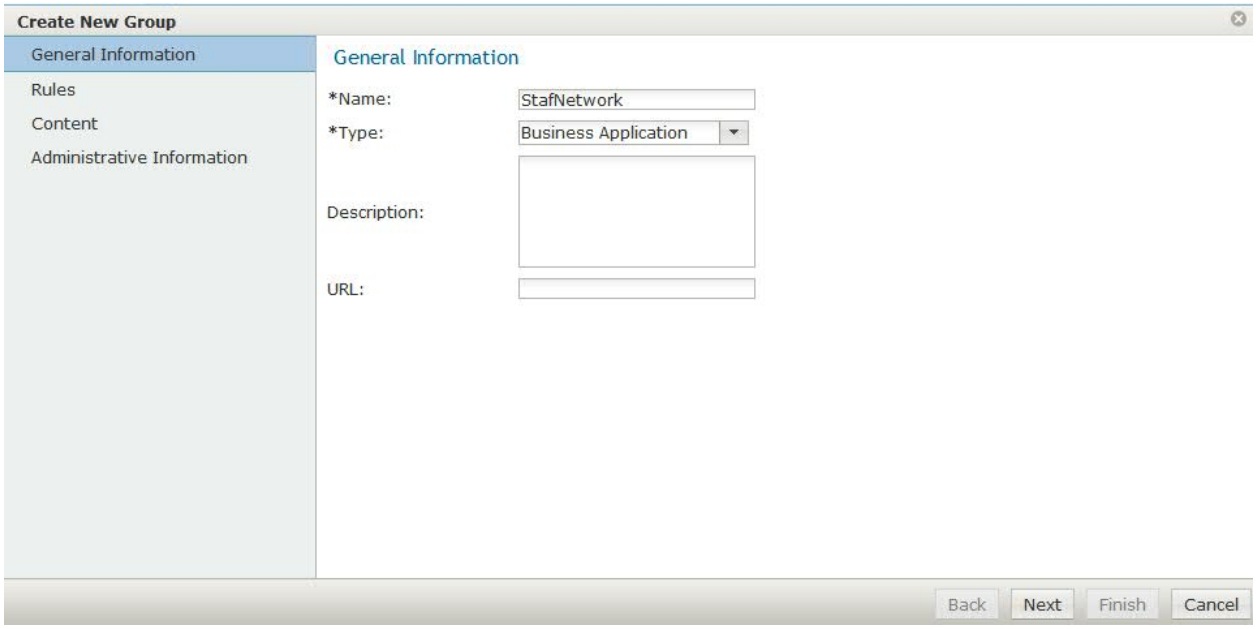


Figure 16 Grouping Composer General Information

6. Click the **Rules** tab.

Complete the following information:

Rule Name: StafNetwork (any name could be used here, in this case both rule and grouping composer object share the same name).

Function Group Name: StafCST (This is the functional group name assigned – any name could be used here).

In the **Select * From** field, enter *com.collation.platform.model.topology.app.AppServer, com.collation.platform.model.discovery.template.AppServerTemplate*

In the **WHERE** field, enter *AppServer.objectType==com.collation.platform.model.discovery.template.AppServerTemplate. name and com.collation.platform.model.discovery.template.AppServerTemplate.type=='AppServer' and not com.collation.platform.model.discovery.template.AppServerTemplate.internal AND (objectType contains "**SysTestAutoFrameworkSTAF**")*

Click ***Save*** then click ***Next***.

Important: ***SysTestAutoFrameworkSTAF*** matches the custom server template (CST) that was created earlier.

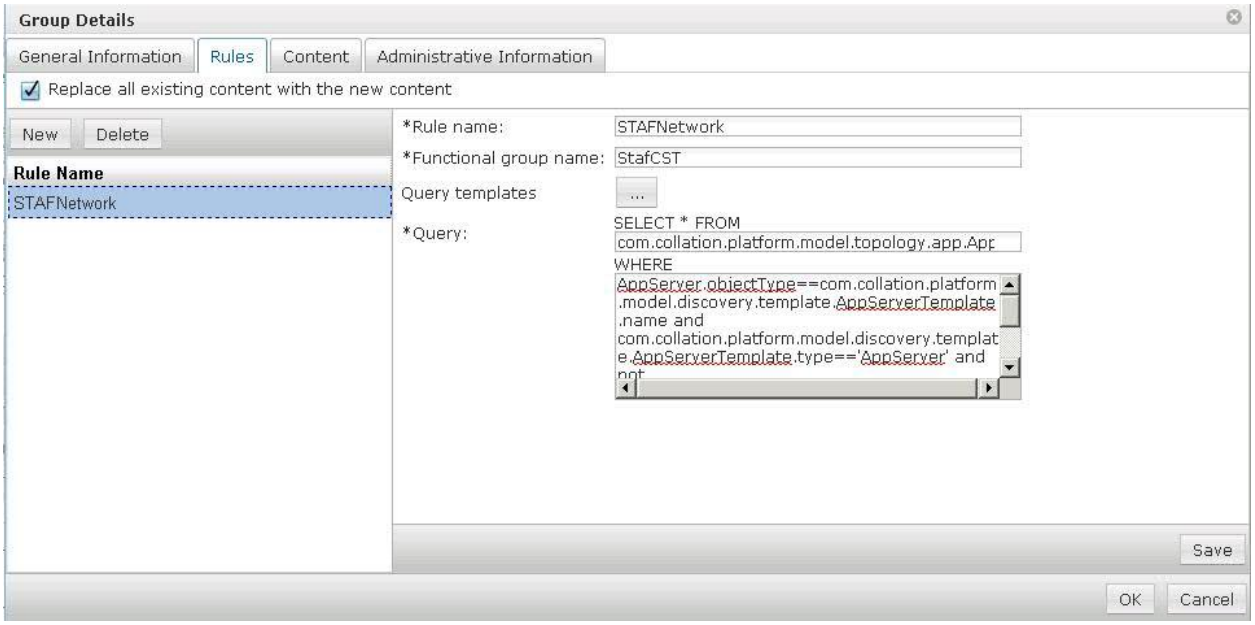


Figure 17 Grouping Composer Rules for STAF

Important: The purpose of the check box on the Rules page is for your grouping composer objects to add and remove content. TADDM usually does not remove components. For more information about this see the section named TADDM: Cleaning up dormant components.

7. On the **Content** tab click **Next**.

8. On the **Administrative Information** tab, click **Finish** to create your group.

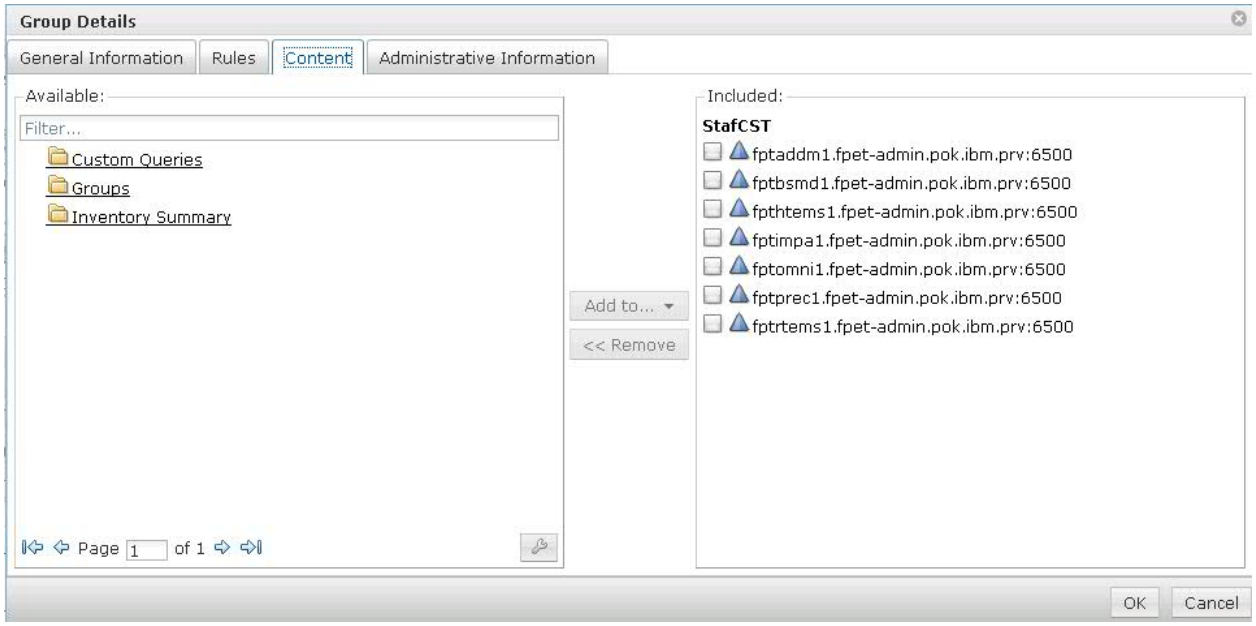


Figure 18 StafNetwork rule-based content

8.1.4 Grouping composer hierarchical example

The following example is not part of the setup for STAF, but rather a standalone hierarchical example.

You can use the grouping composer to create a hierarchical business service by first creating business applications for the “contained” configuration instances. In the following example, Dorfman1.1.1 and Dorfman1.1.2 business applications have been created. Using the grouping composer, you now create the Dorfman1.1 business service which contains these two business applications. Again, it is recommended that you do this in TBSM, but TADDM does provides this capability.

1. On the general information page, enter `Dorfman1` in the **Name** field and click **Next**.

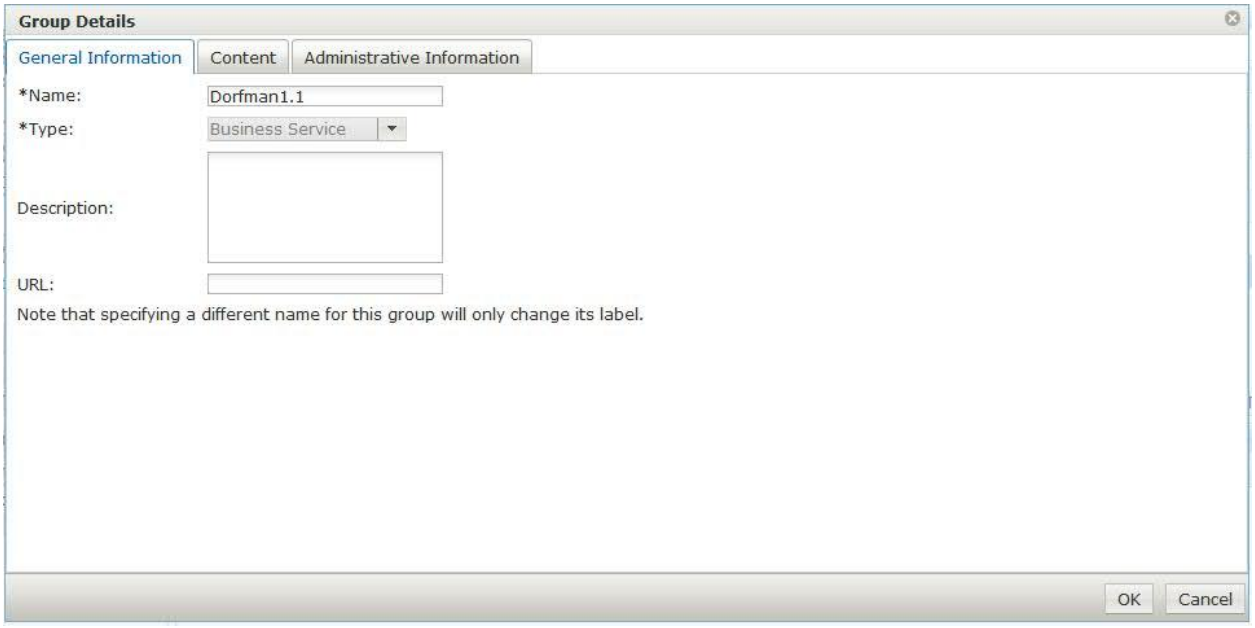


Figure 19 Grouping Composer for Hierarchy

2. On the Content page, click **Groups > Business Application**

3. Under the Available list, select the **Dorfman1.1.1** and **Dorfman1.1.2** check boxes. Click **Add** to move these two items to the **Included** list and click **Next**. Then click **Finish**. Results are shown in Figure 20 Creating a Business Service containing Business Applications.

In this same way, you can create grouping composer objects (both business applications and collections) without using rules. You do this by selecting objects that are known to TADDM from the Content page of the grouping composer. Note that business services can only be defined statically, as they are created using the Content page.

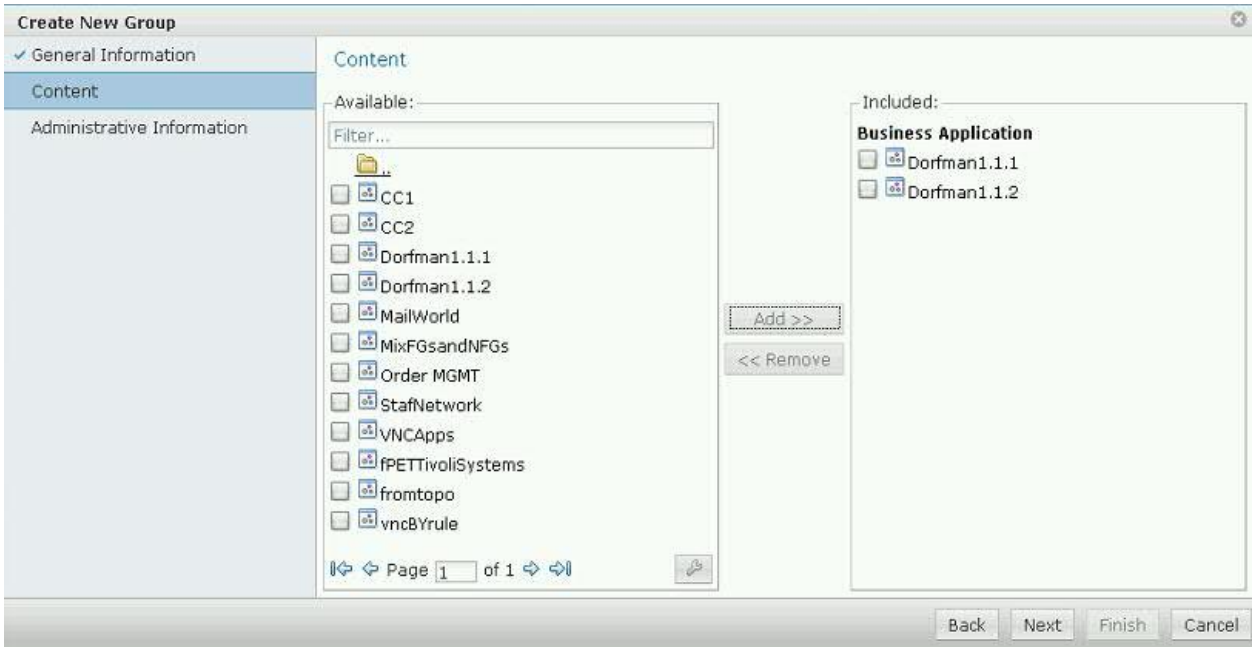


Figure 20 Creating a Business Service containing Business Applications

The previous steps explained how to create a hierarchical structure using the grouping composer. Figure 20 is shows what will be displayed on the TBSM Service Configuration or Service Navigation portlet when displaying Imported Business Services.

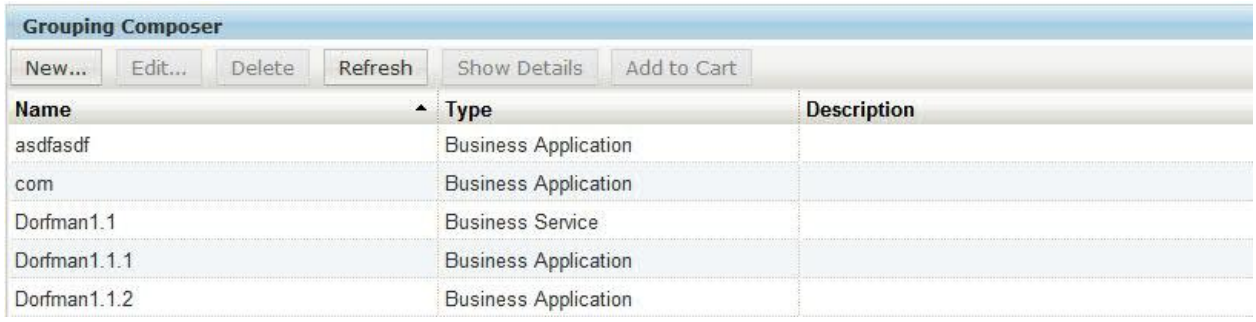


Figure 21 TBSM Display of Business Service Hierarchy

Business applications and business service names can also include other characters, including the following characters:

(space) ( ) - .

The following example shows additional characters in the name of grouping composer created object as displayed in either TADDM or TBSM.

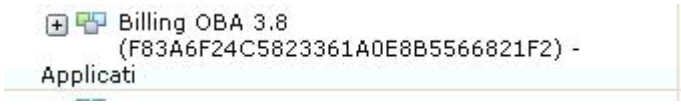


Figure 22 More Complex Naming Example

Avoid these characters in TADDM Group Names

TADDM allows a wider set of characters than does TBSM. If creating groups for integration with TBSM do not use the following characters in the **Name** field:

* ; < = > ? \ |

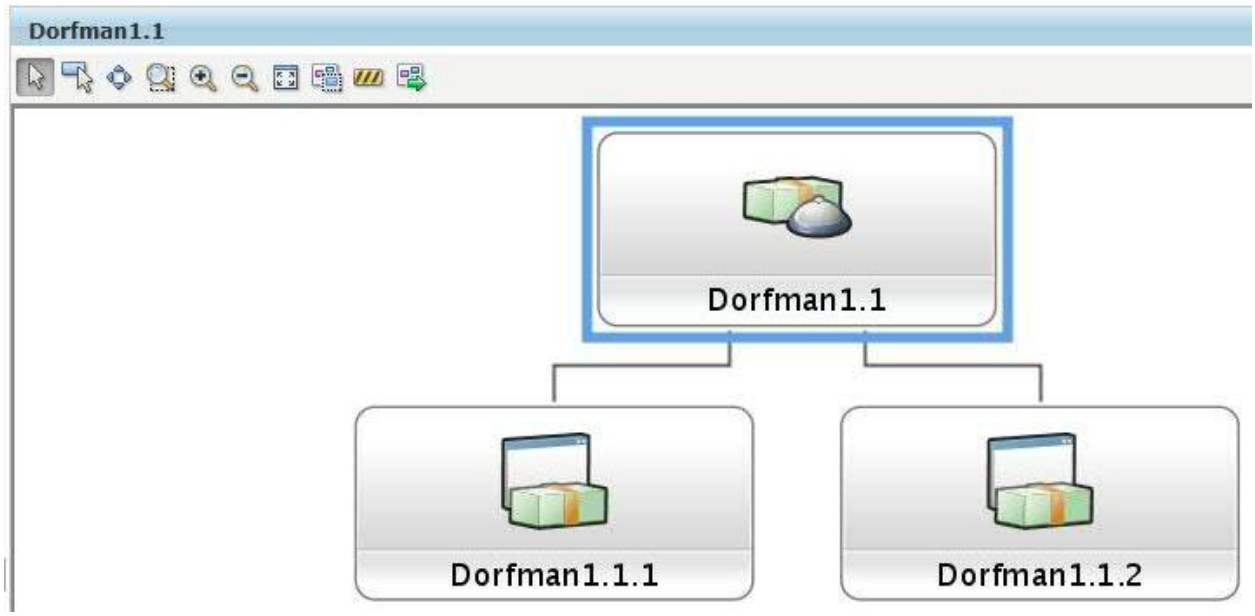


Figure 23 Hierarchy Topology in TADDM

8.2 TBSM Configuration

This section includes the TBSM configuration steps required to complete this solution.

8.2.1 Editing the rules file for custom applications

The toolkit transfers business applications, business services, and other *mapped* configuration items from TADDM without changes. However, the custom server template previously added does require a change. To transfer a custom application, edit the following configuration file in the Discovery Library Toolkit:

In %tbsmhome%/XMLtoolkit/xml edit the CDM_TO_TBSM4x_MAP.xml file.

Add a `<condition...>` and `</condition>` for each custom server (application) template that you created in TADDM to the CustomAppl rule. The three added here are not indented to highlight the additions. Note that "StafCST" is one of the three shown below.

```
<Rule name="CustomAppl">
  <condition attrKeyword='CdmObjectType' operator="=" value='BroadVision'>
    <condition booleanOperator='OR' operator="=" value='Quadstone'>
      <condition booleanOperator='OR' operator="=" value='Microsoft BizTalk'>
        <condition booleanOperator='OR' operator="=" value='OracleStrayProcesses'>
          <condition booleanOperator='OR' operator="=" value='Netegrity-Siteminder'>
            <condition booleanOperator='OR' operator="=" value='Remedy ARS'>
              <condition booleanOperator='OR' operator="=" value='RIM BlackBerry'>
                <condition booleanOperator='OR' operator="=" value='SiebelServer'>
                  <condition booleanOperator='OR' operator="=" value='SiebelGateway'>
                    <condition booleanOperator='OR' operator="=" value='SiebelGateway'>
                      <condition booleanOperator='OR' operator="=" value='Symantec Anti-virus Agent'>
                        <condition booleanOperator='OR' operator="=" value='Microsoft HyperV'>
<condition booleanOperator='OR' operator="=" value='PRMail'>
<condition booleanOperator='OR' operator="=" value='SysTestAutoFrameworkSTAF'>
<condition booleanOperator='OR' operator="=" value='VNCserver'>
</condition>
</condition>
</condition>
                      </condition>
                    </condition>
                  </condition>
                </condition>
              </condition>
            </condition>
          </condition>
        </condition>
      </condition>
    </condition>
    <Component/>
  </Rule>
```

Figure 24 Adding Custom Application support for DLT

8.2.2 Running the DLT

Run the TBSM DLT to transfer TADDM objects including the " StafCST" custom server template and the business application created with the grouping composer entitled "StafNetwork".

To run the DLT go to the %TBSM_HOME%/XMLtoolkit/bin directory and issue the following command:

./cmdbdiscovery.sh –r –b

The "-b" option should be used on the first pull or anytime a complete resynchronize is needed.

8.2.3 Dashboard pre-configuration

To create a dashboard for the STAF application, first create a data source, data fetcher, template and a service for STAF.

8.2.4 Add data source for PostgresDB

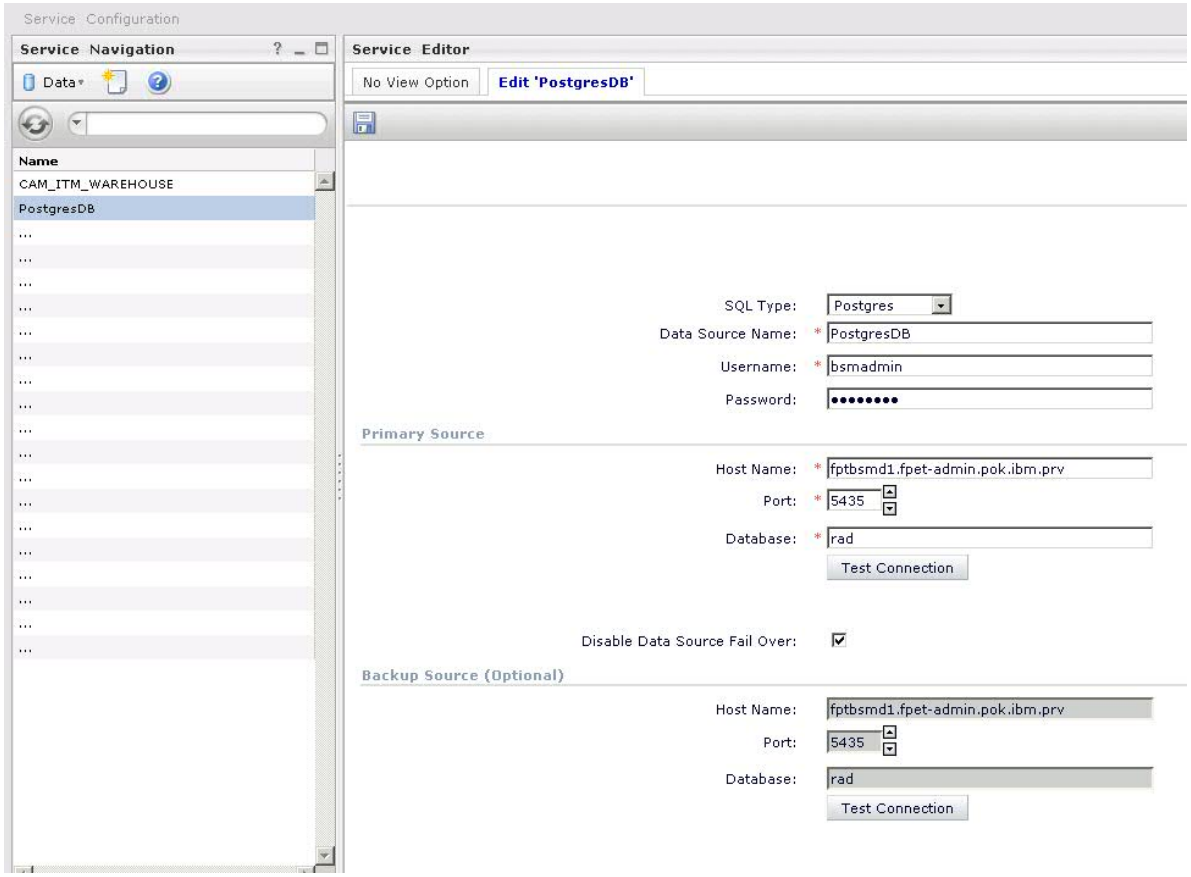


Figure 25 Data Source Creation

8.2.5 Adding a Data Fetcher

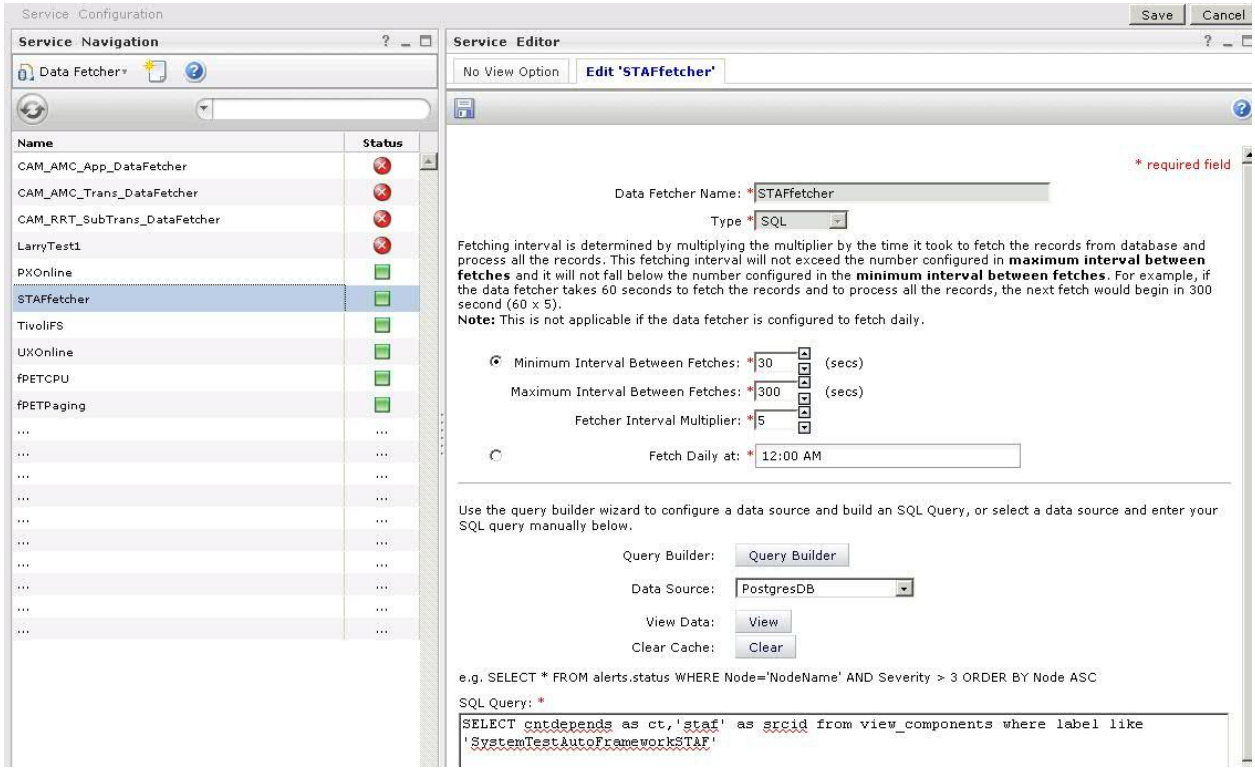

Service Configuration
Save
Cancel
Service Navigation
Data Fetcher
Name
Status
CAM_AMC_App_DataFetcher
CAM_AMC_Trans_DataFetcher
CAM_RRT_SubTrans_DataFetcher
LarryTest1
PXOnline
STAFfetcher
TivoliFS
UXOnline
fPETCPU
fPETPaging
Service Editor
No View Option
Edit 'STAFfetcher'
* required field
Data Fetcher Name: *STAFfetcher
Type *SQL
Fetching interval is determined by multiplying the multiplier by the time it took to fetch the records from database and process all the records. This fetching interval will not exceed the number configured in maximum interval between fetches and it will not fall below the number configured in the minimum interval between fetches. For example, if the data fetcher takes 60 seconds to fetch the records and to process all the records, the next fetch would begin in 300 second (60 x 5).
Note: This is not applicable if the data fetcher is configured to fetch daily.
Minimum Interval Between Fetches: *30 (secs)
Maximum Interval Between Fetches: *300 (secs)
Fetcher Interval Multiplier: *5
Fetch Daily at: * 12:00 AM
Use the query builder wizard to configure a data source and build an SQL Query, or select a data source and enter your SQL query manually below.
Query Builder:
Query Builder
Data Source:
PostgresDB
View Data:
View
Clear Cache:
Clear
e.g. SELECT * FROM alerts.status WHERE Node='NodeName' AND Severity > 3 ORDER BY Node ASC
SQL Query: *
SELECT cntdepends as ct,'staf' as srcid from view_components where label like 'SystemTestAutoFrameworkSTAF'

Figure 26 Data Fetcher Query for Application

STAFfetcher SQL Query is set as follows: Select cntdepends as ct,'staf' as srcid from view_components where label like 'SystemTestAutoFrameworkSTAF'. This query uses the data source created in the previous step to obtain the number of staf custom servers running in the environment. This count along with srcid set to 'staf' gives us the two pieces of information you use in the template and service to obtain the count. The results of the query are shown after clicking the View button shown in Figure 26 Data Fetcher Query for Application. The results are shown in Figure 27 View Query Results.

8.2.6 Data fetcher view results for staf custom server

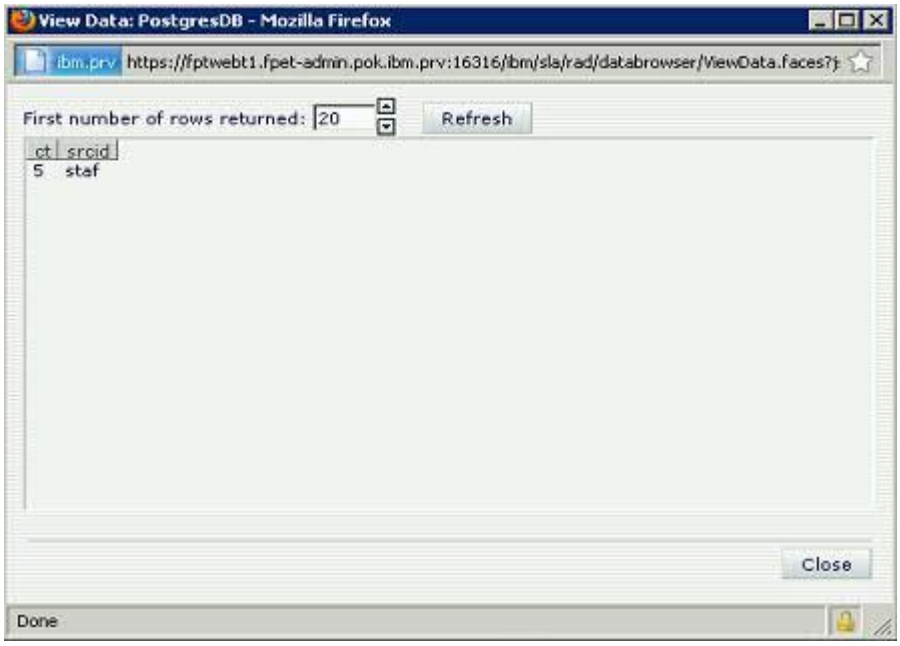


Figure 27 View Query Results

8.2.7 Create a template to be used by the Dashboard Service Portlet

From the service navigation page for Templates, click on the new template icon highlighted below and create Template shown. Template name and description are set to “STAFRule”. Then click on the incoming status rule image, also highlighted below.

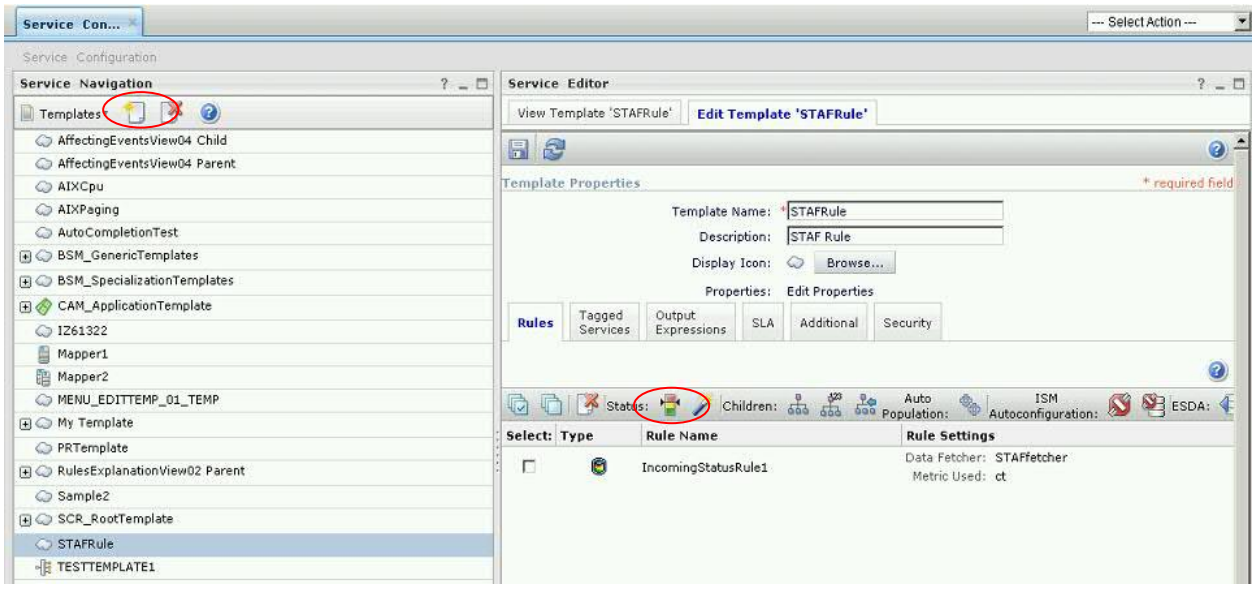


Figure 28 Create a Template

Incoming status rule shown above must be added. Take the defaults on the first page and click OK.

Figure 29 Incoming status rule selector

Take the default of IncomingStatusRule1 for the rule name. For data feed select STAFfetcher you created in previous steps. Configure the rest of this page as shown in the following image:

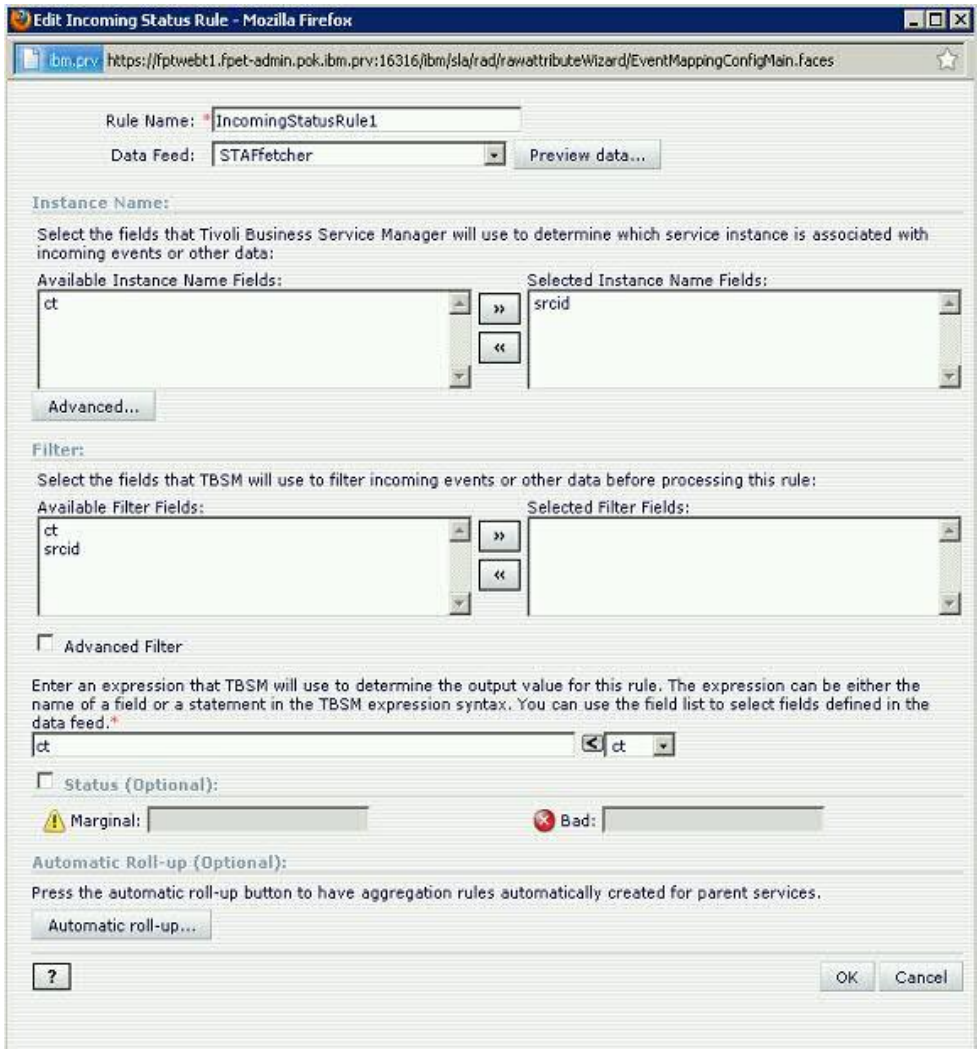

Edit Incoming Status Rule - Mozilla Firefox
https://fptwebt1.fpet-admin.pok.ibm.prv:16316/ibm/sla/rad/rawattributeWizard/EventMappingConfigMain.faces
Rule Name: IncomingStatusRule1
Data Feed: STAFfetcher
Preview data...
Instance Name:
Select the fields that Tivoli Business Service Manager will use to determine which service instance is associated with incoming events or other data:
Available Instance Name Fields:
ct
Selected Instance Name Fields:
srcid
Advanced...
Filter:
Select the fields that TBSM will use to filter incoming events or other data before processing this rule:
Available Filter Fields:
ct
srcid
Selected Filter Fields:
Advanced Filter
Enter an expression that TBSM will use to determine the output value for this rule. The expression can be either the name of a field or a statement in the TBSM expression syntax. You can use the field list to select fields defined in the data feed.*
ct
Status (Optional):
Marginal:
Bad:
Automatic Roll-up (Optional):
Press the automatic roll-up button to have aggregation rules automatically created for parent services.
Automatic roll-up...
OK
Cancel

Figure 30 Incoming status rule

8.2.8 Create a service which uses this template

When creating the portlets in the next session you need a service tree parent and child to contain our service template. From the service navigation services menu, create a parent and child service entry. Click the new service icon highlighted in the upper left of the image below. Set the service name to TestSub (this is the child service) and assign it template you created in the previous steps “STAFRule”. Then click the save icon.

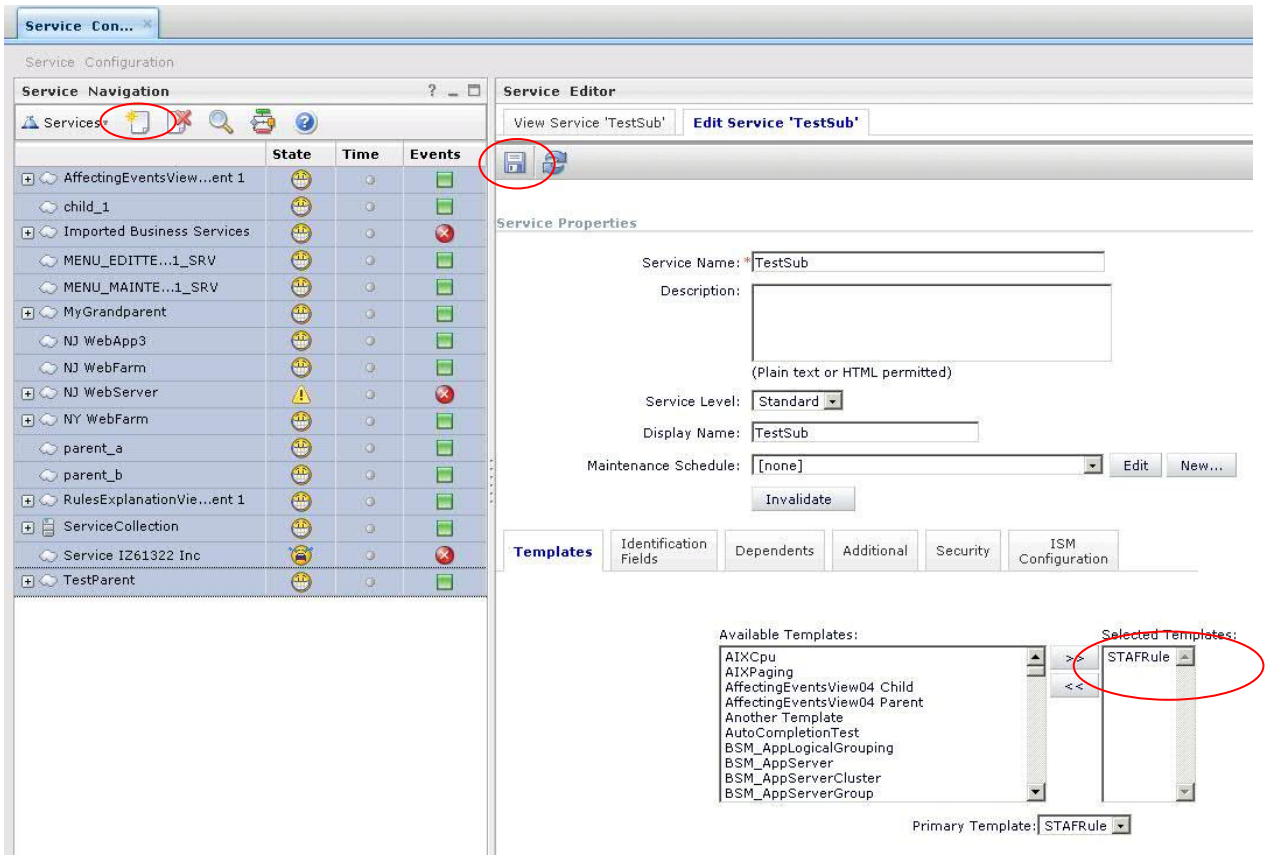


Figure 31 TestSub uses the STAFRule Template

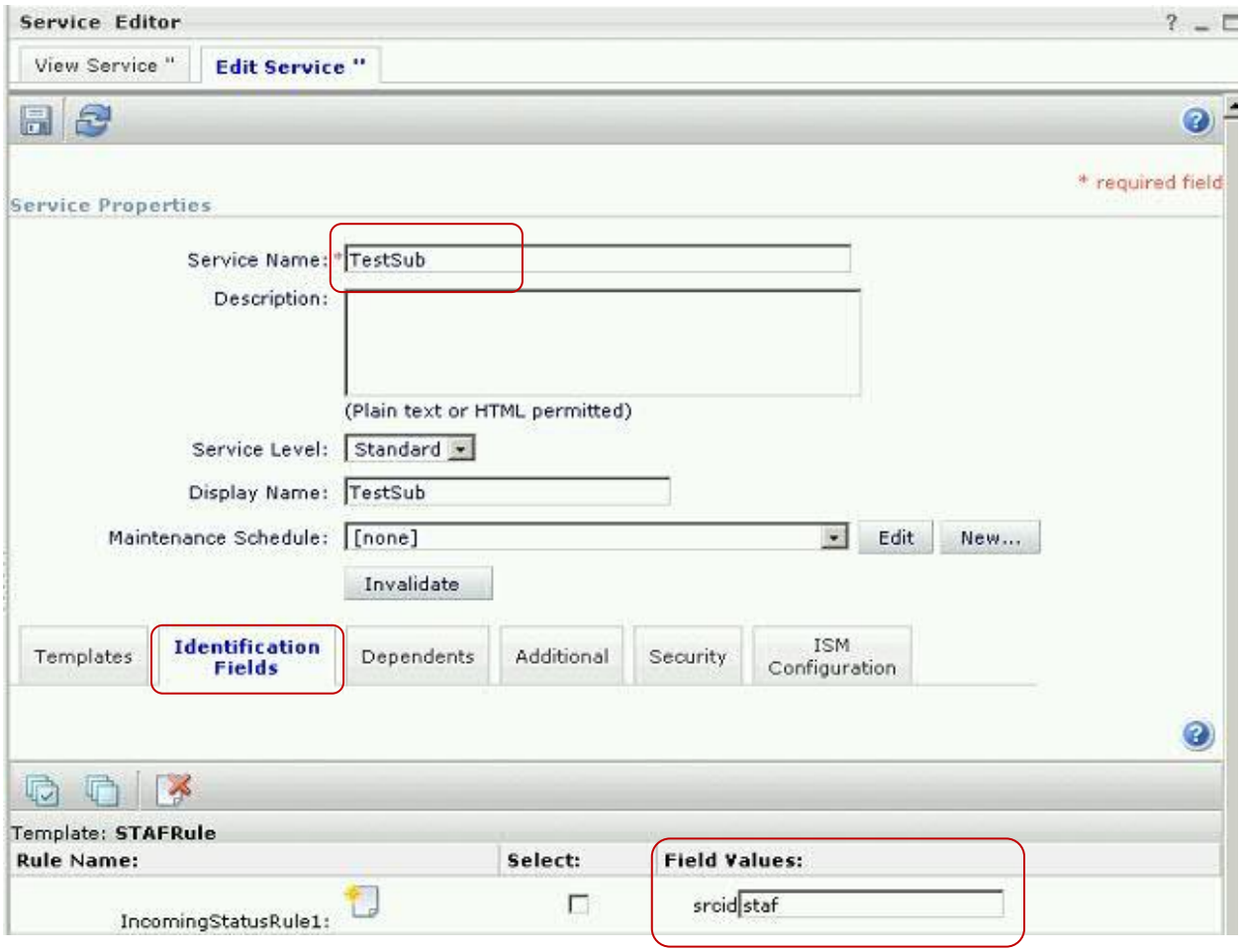


Figure 32 Set scrid filter for incoming status rule

The TestSub service contains the template created in Figure 28 Create a Template. It then filters the data based on srcid, in this example srcid of "STAF" is associated with the count of STAF servers (ct). Review Figure 26 Data Fetcher Query for Application for more details.

Set the service name to TestParent (this is the parent service) and set its dependent to the "TestSub" (the child service created above). Then save this service.

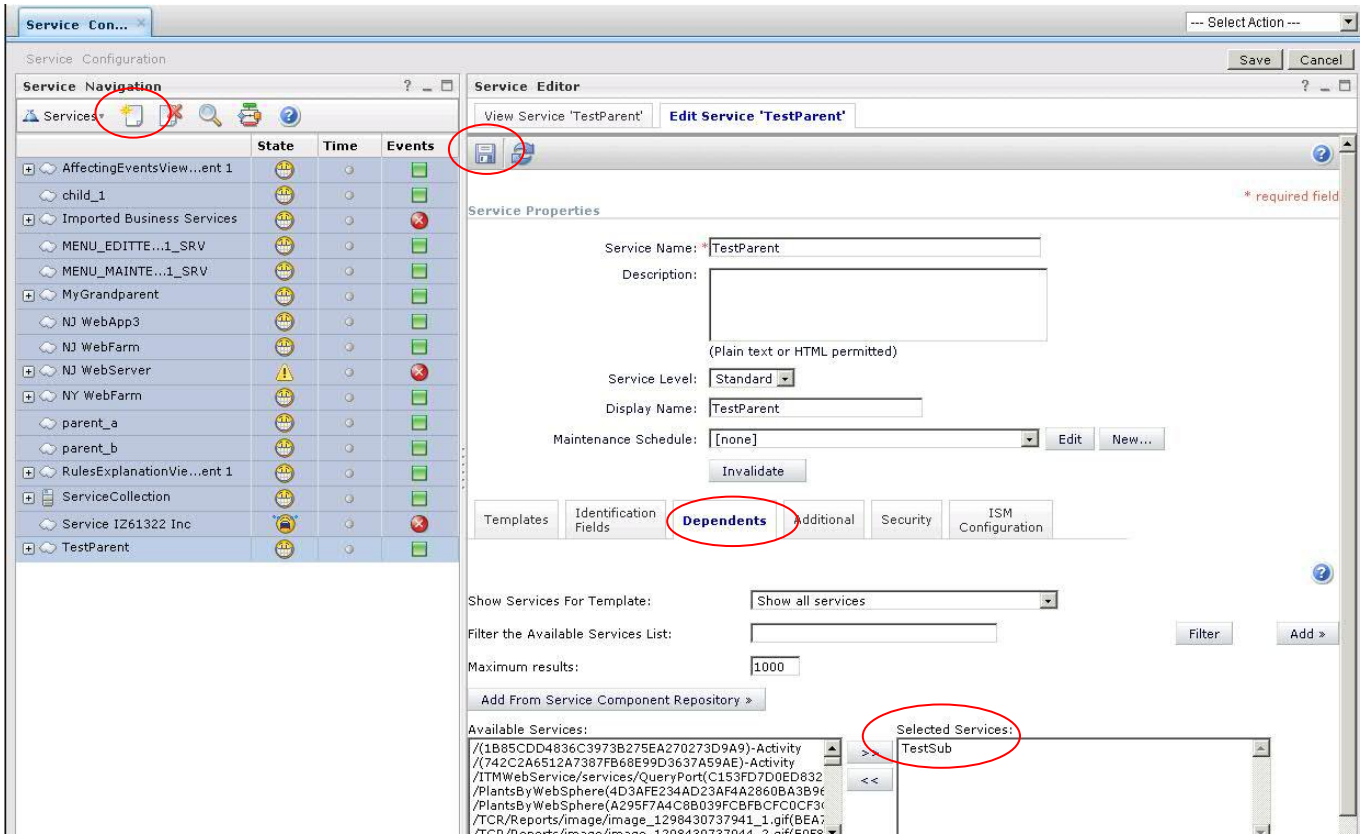


Figure 33 Create Parent object for the dashboard portlet service

In the figure above you create a new service "parent" object and set the Dependent(s) to `TestSub.`

The service tree now contains these services. TestSub is used later to create the portlet also shown below.

Figure 34 TBSM tree view of the dashboard service

8.3 Portlet Creation Step by Step

Now that all pre-requisite pieces are in place. It is time to create the dashboard for the STAF service. The steps shown in this section create the lower left widget below and the image shown in Figure 10 Sample dashboard.

Figure 35 Widget portion of the dashboard

8.3.1 Create a new page

From `Settings > Page Management` click `New Page`.

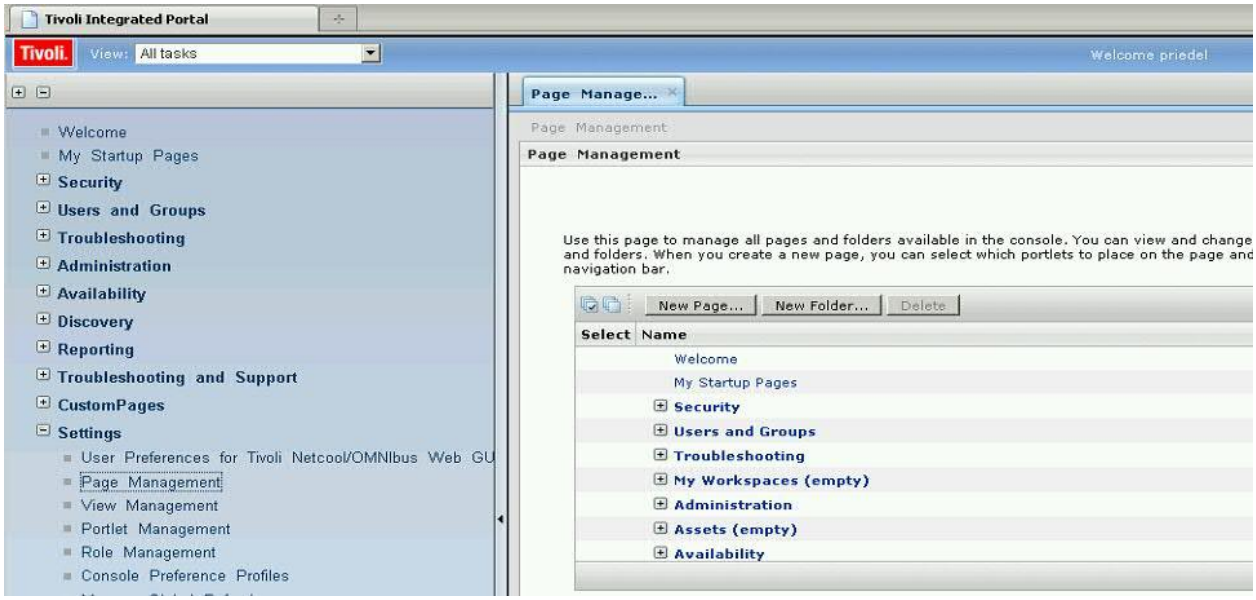


Figure 36 Create a new custom page

8.3.2 Set portlet type to Service Tree

From the new page menu select the Service Tree portlet entity type. Click OK and then Save.

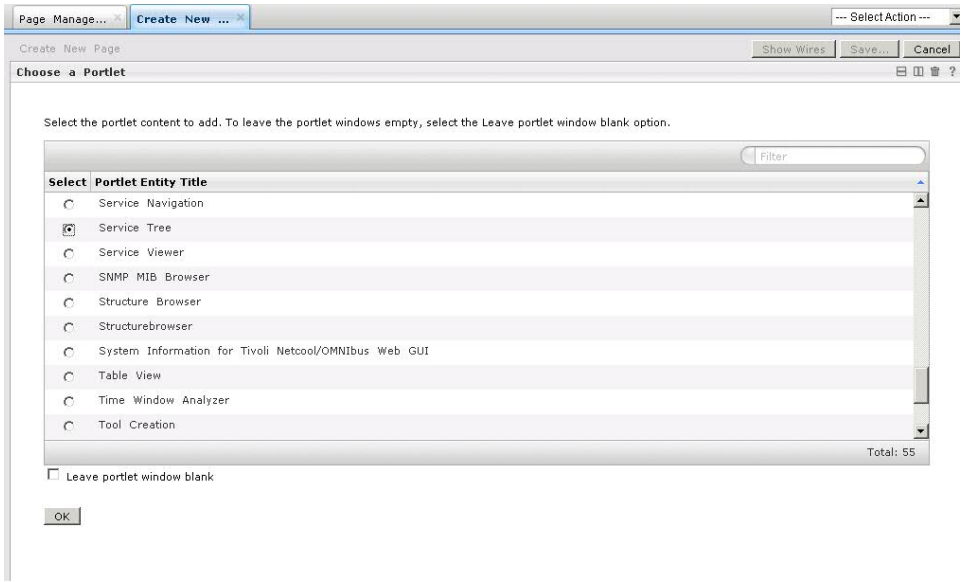


Figure 37 Select the Service Tree Portlet

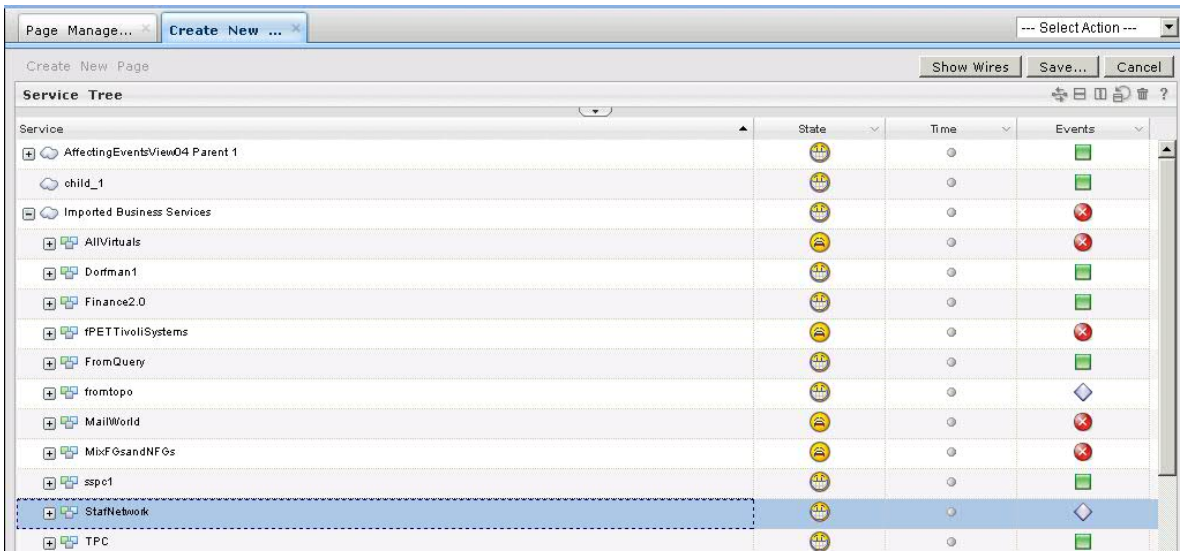


Figure 38 View of new portlet

8.3.3 Save the page with name “staf-status”

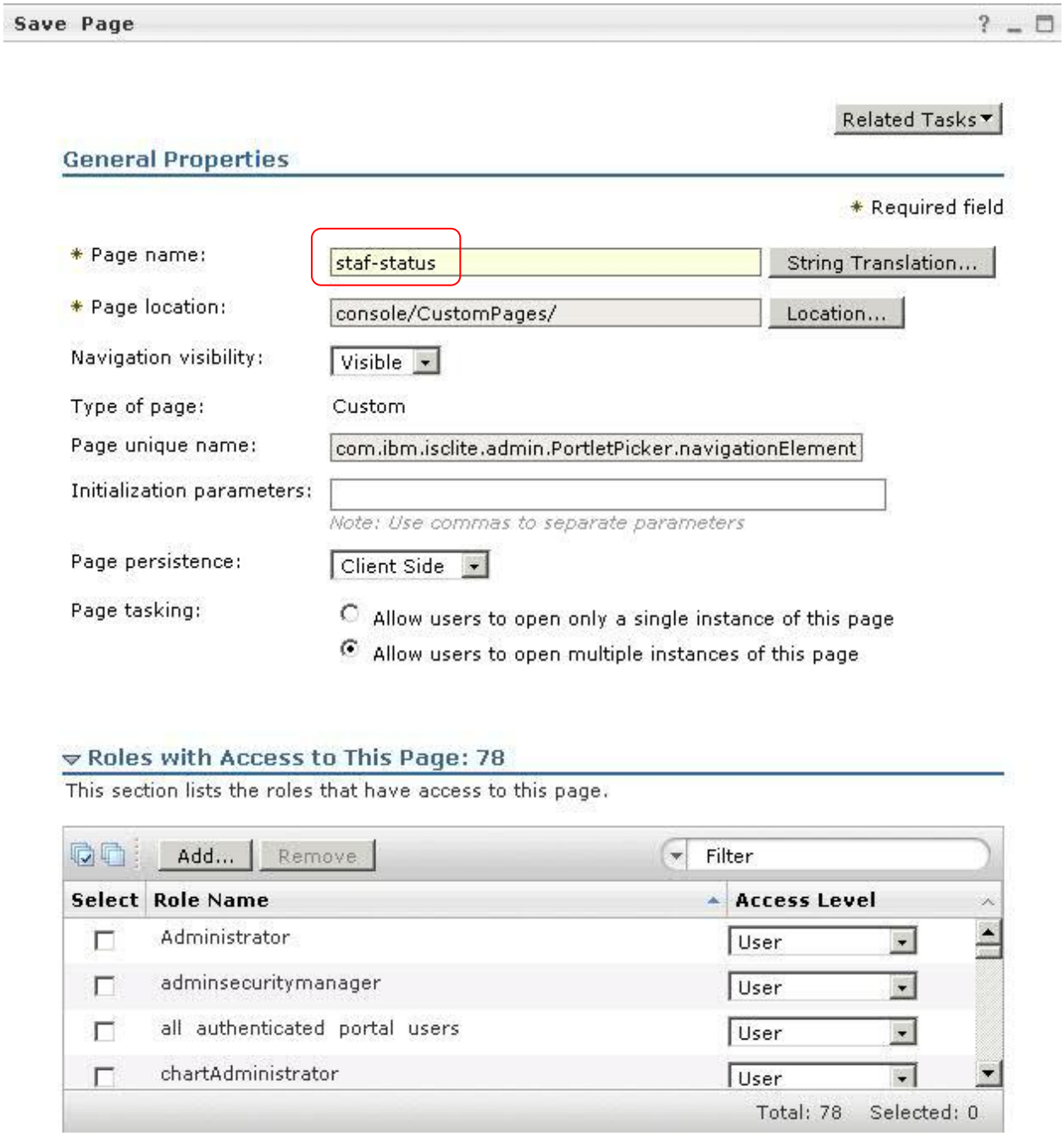


Figure 39 Saving the new portlet page

When you create a new page, assign the operator roles and access levels as deemed appropriate. Consult your security administrator for more details.

8.3.4 Edit the defaults for your new Service Tree

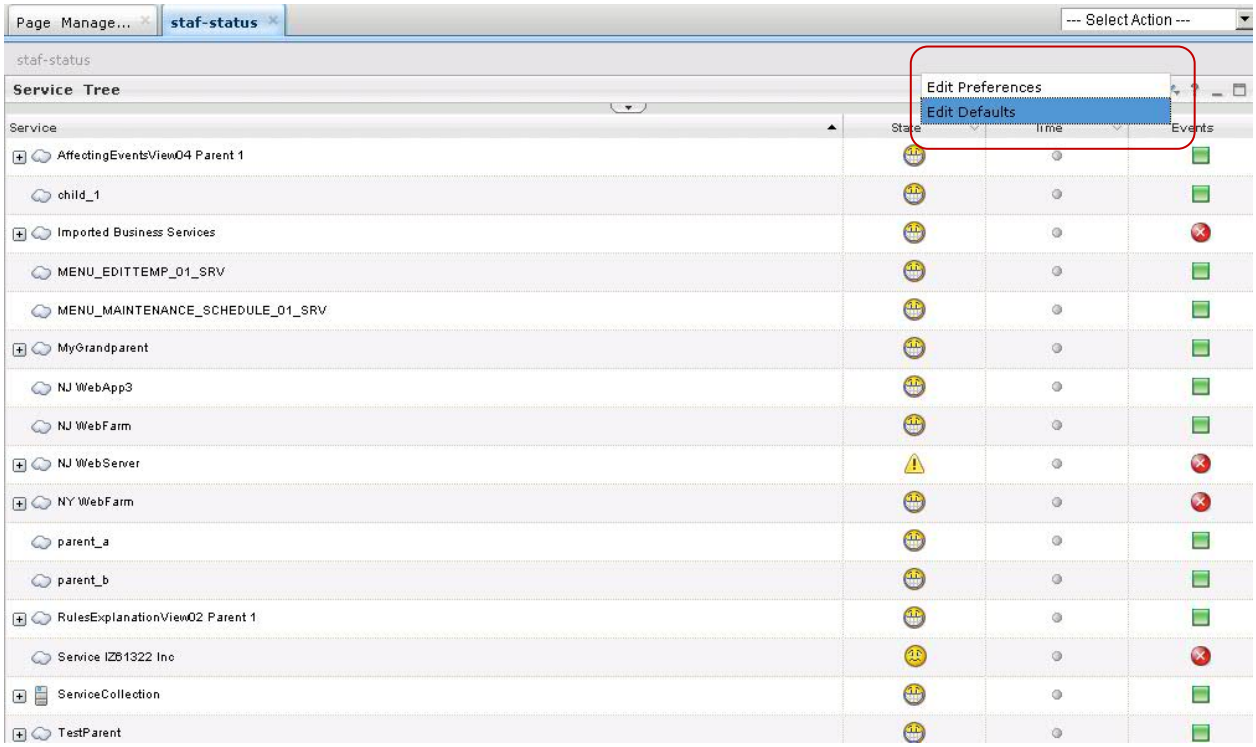


Figure 40 Defaults Edit

> **Preferences versus Defaults**
>
> Defaults set values for all uses of the portlet unless overridden by preferences. Set preferences only if you want to override the defaults.

You need to edit the defaults (and/or preferences) to this page to set the what portion of the service tree applies to “staf-status”.

8.3.5 Set starting instance to Business Application (STAF)

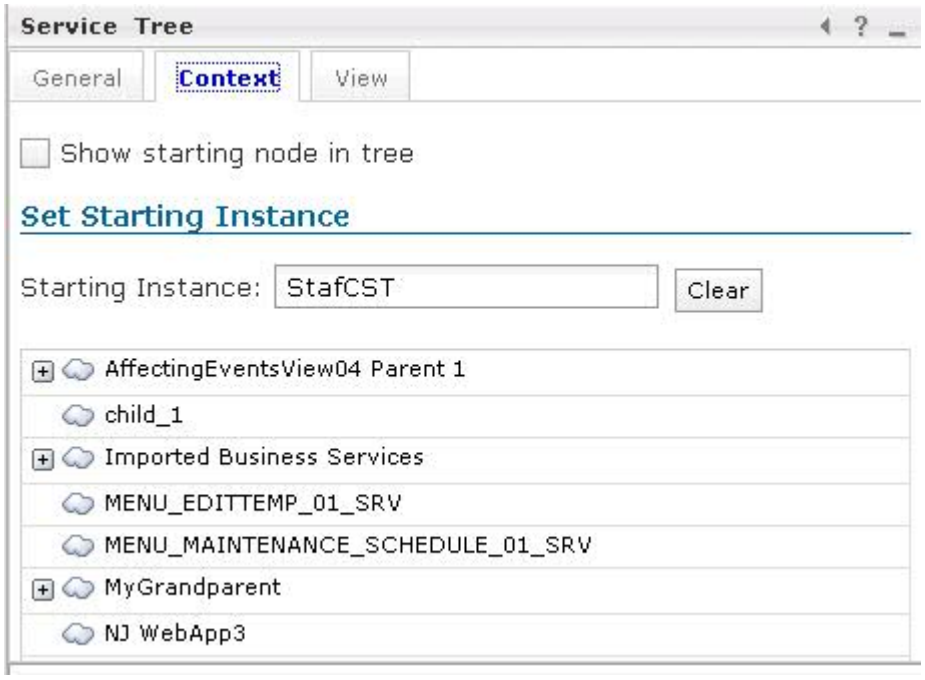


Figure 41 Set STAF as the starting instance

StafCST is the functional group name set in the TADDM grouping composer for this example. To find it, click on Imported Business Services to expand the list and then click and expand the imported business application entitled

8.3.6 Save your changes between steps

Figure 42 Save the changes – found in the lower left corner

8.3.7 Edit Page to add additional portlets

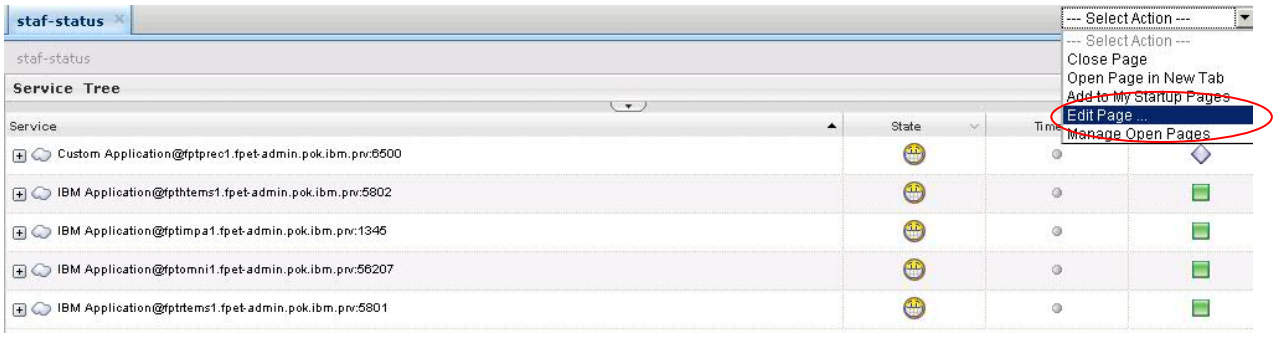


Figure 43 Re-edit the page to create additional portlets

8.3.8 Perform a vertical split

Figure 44 Vertical split

Split the current portlet vertically as shown above. This results in 2 portlets on the custom page being built.

8.3.9 Choose AEL Portlet

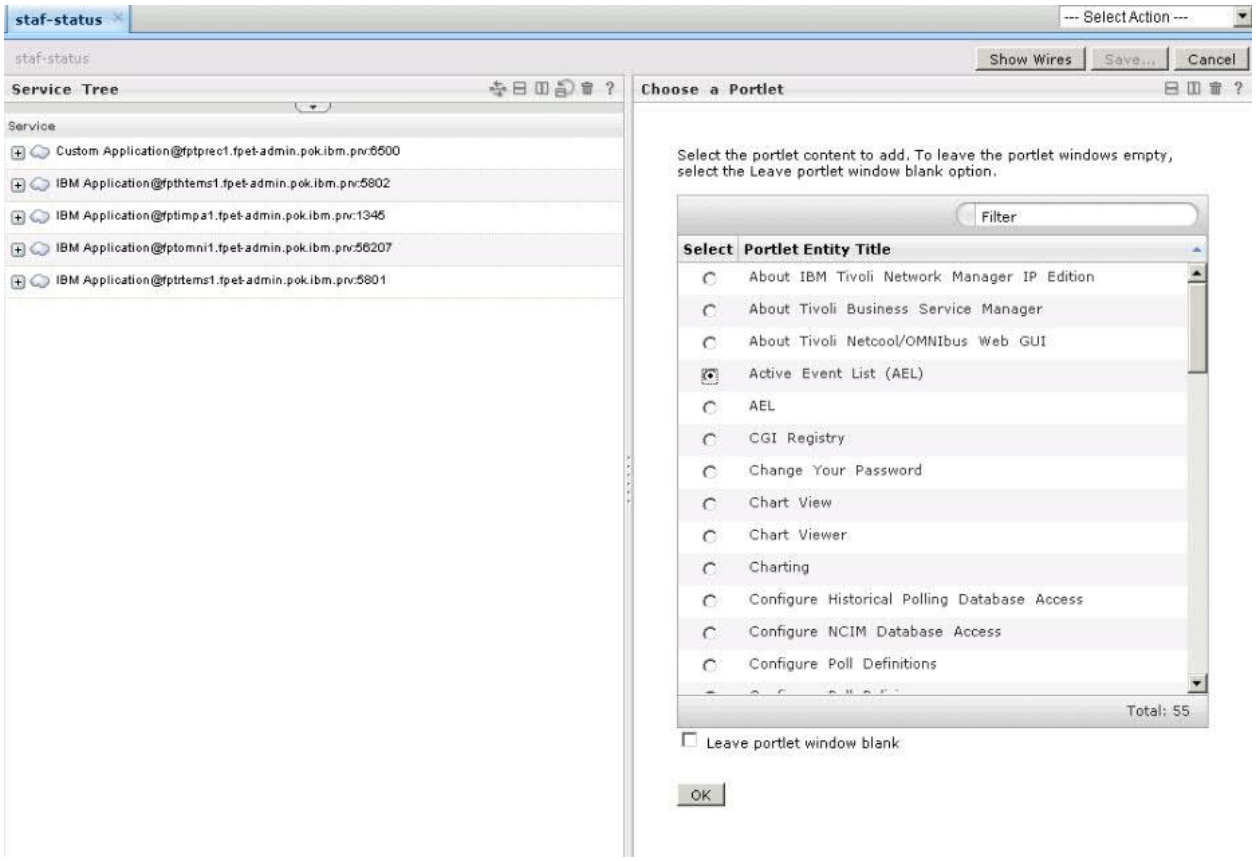


Figure 45 Choose a portlet type

For the upper right portlet, select “Active Event List (AEL)” portlet, here you display the events related to the services selected in the service tree portlet. TBSM can display and correlate events of the Omnibus Object Server. In this solution the source of the events comes from the IBM Tivoli Monitoring product. The events are created as a result of operating system agents and situations. Custom situation for this solution is discussed later in this guide.

8.3.10 Split both portlets Horizontally

To complete the sample dashboard you need to split the both existing portlets horizontally now. Click the highlighted icon as shown in the next figure:

For lower right and left portlets set both as service viewers. As shown in the lower right portlet.

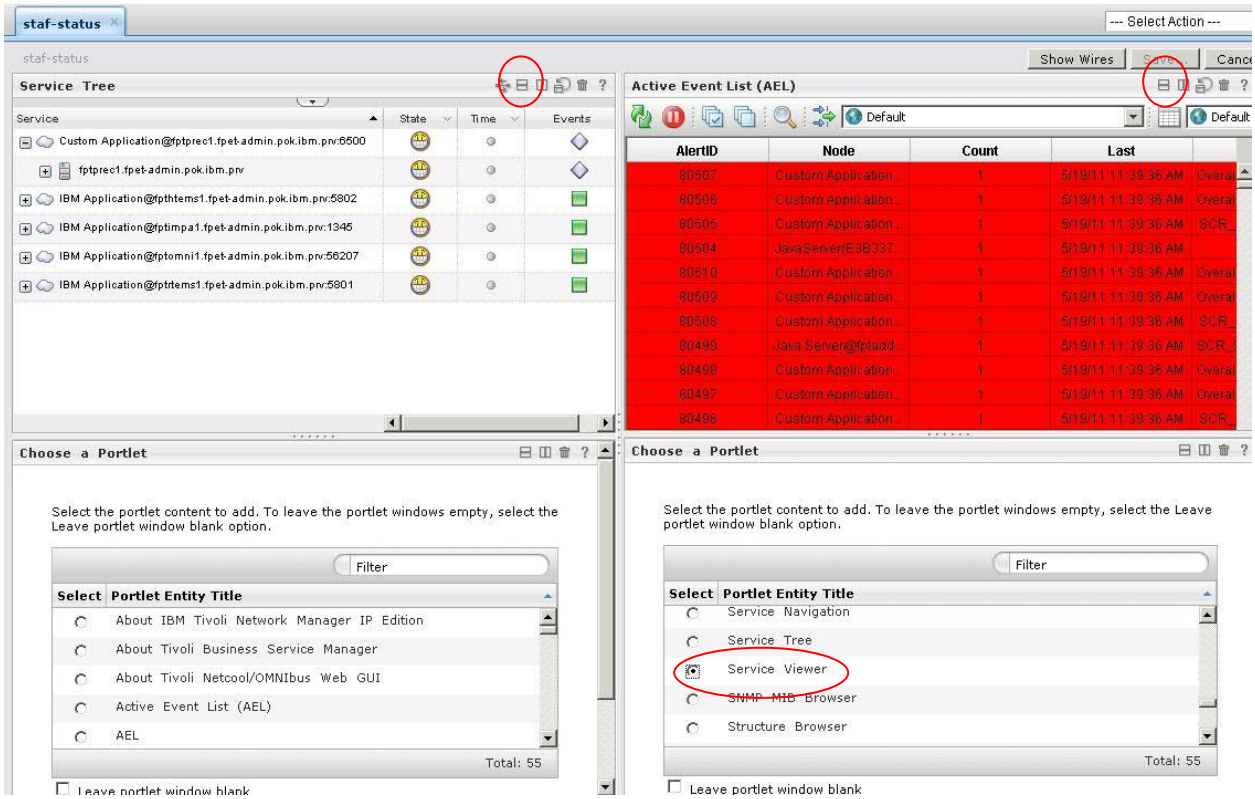
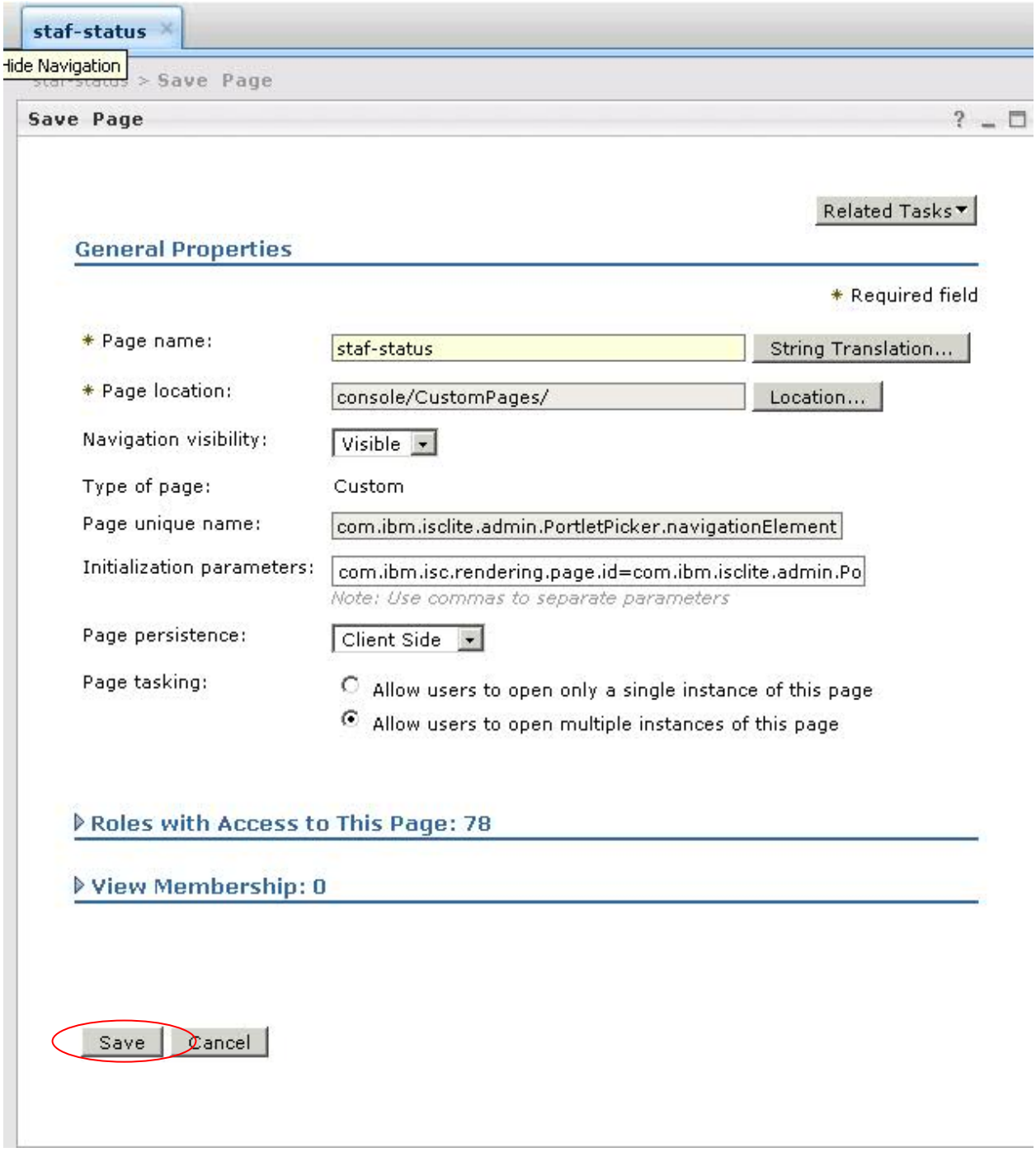


Figure 46 Assign portlet types to the new portlets

8.3.11 Save the new page again



8.3.12 Set context default preferences for lower right to the StafCST imported service

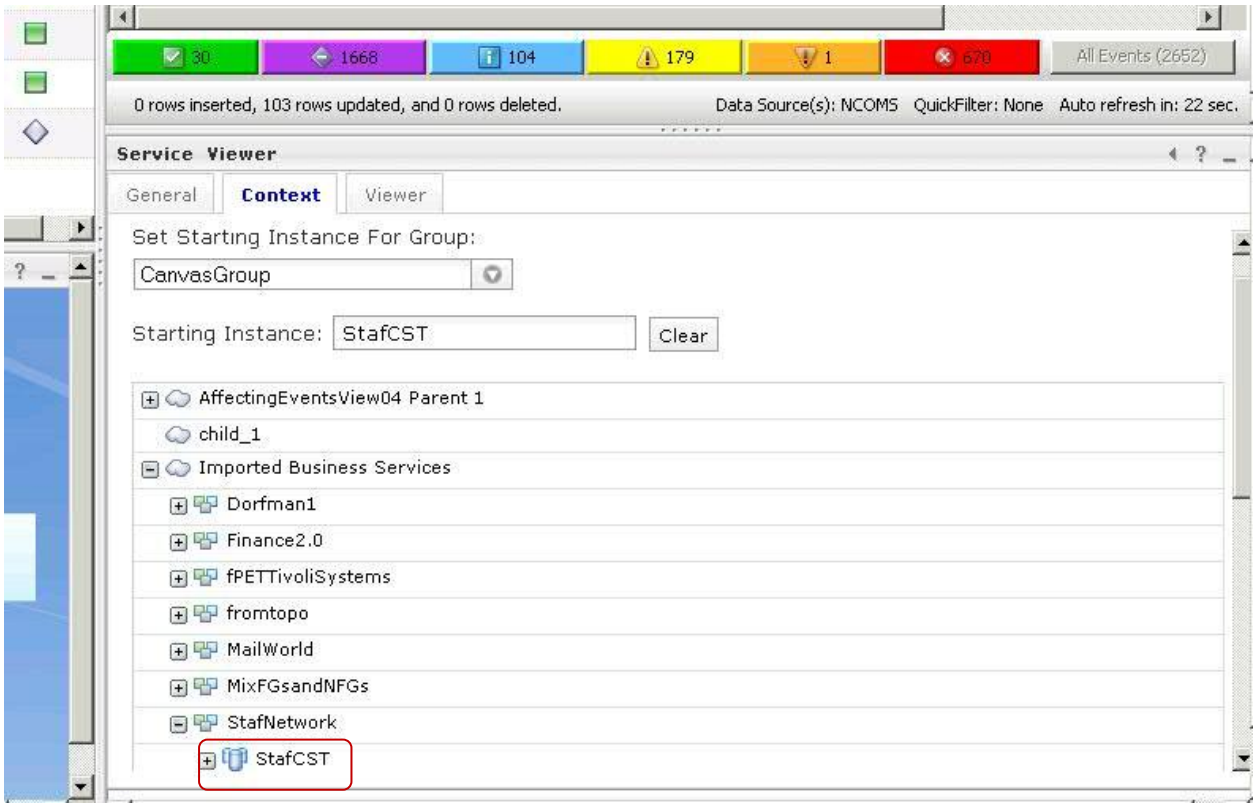


Figure 47 Service Viewer for StafCST

8.3.13 Select viewer type to “Basic Relationships”

This selection combined with the override settings of 1 up and 3 down displays a tree view for StafCST in the lower right portlet.

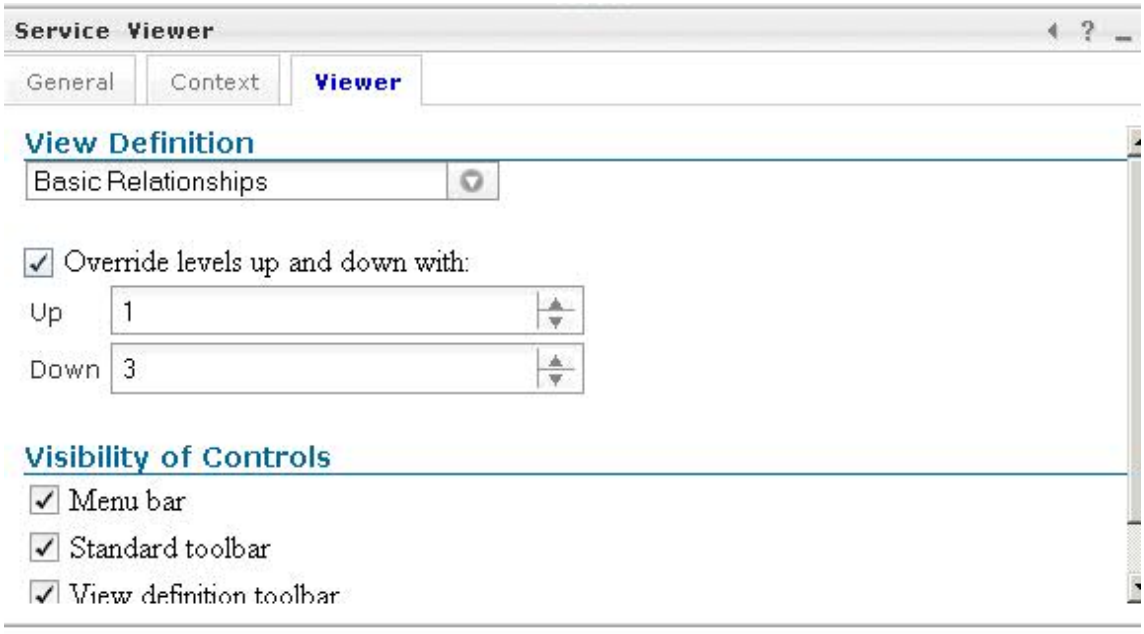


Figure 48 Basic Relationships Viewer

8.3.14 Select lower left context to TestSub

“TestSub” is the service created earlier to track the number of STAF services in our enterprise.

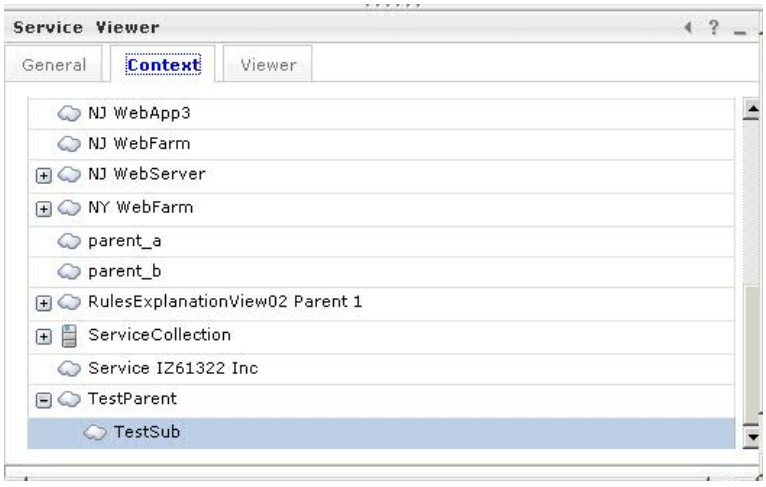


Figure 49 Dashboard service portlet settings

8.3.15 Lower Left portlet viewer type set to “Relationships”

As this portlet is used to display an indicator or widget, it is not important what view definition is selected in this step.

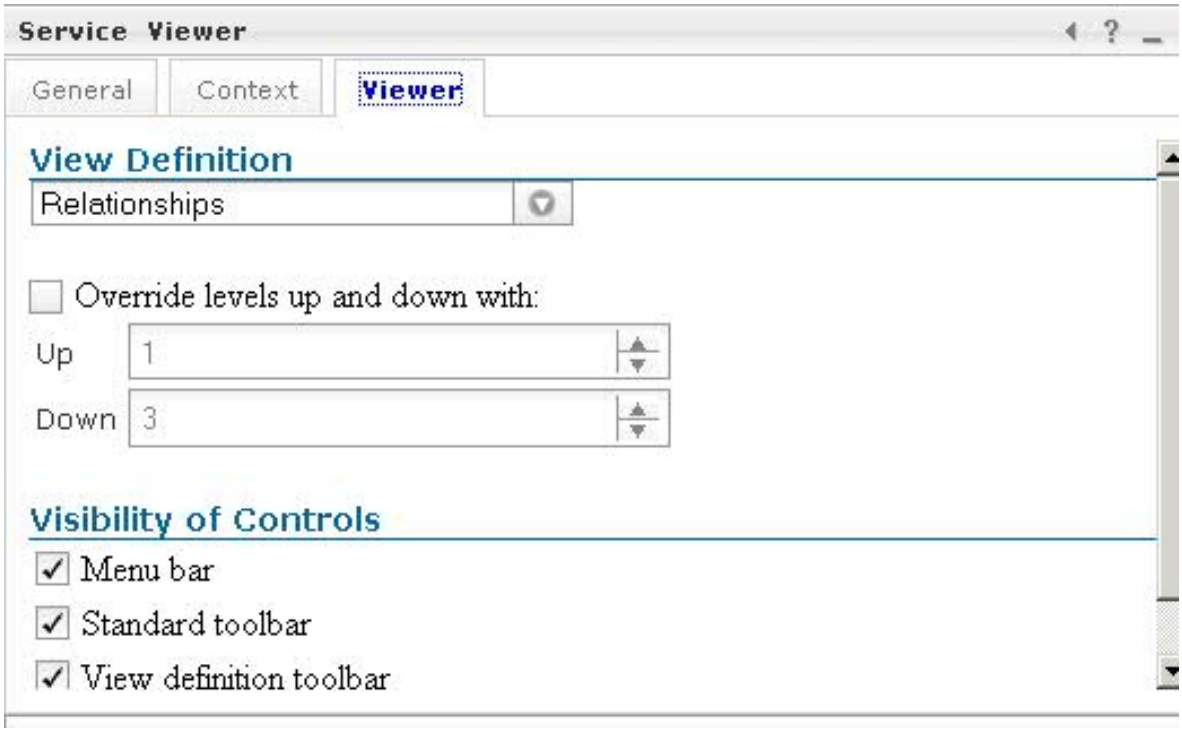


Figure 50 Service Viewer Settings

Terminology

Indicators, gauges and widgets can refer to different object variations in TBSM. In this solution all of these items are grouped as those objects found under the indicators tab.

8.3.16 Remove all wires

For this dashboard you want to remove all “default” wires. While editing the page select “show wires”. Uncheck all the check boxes in the “Enable” column and “Save” the page.

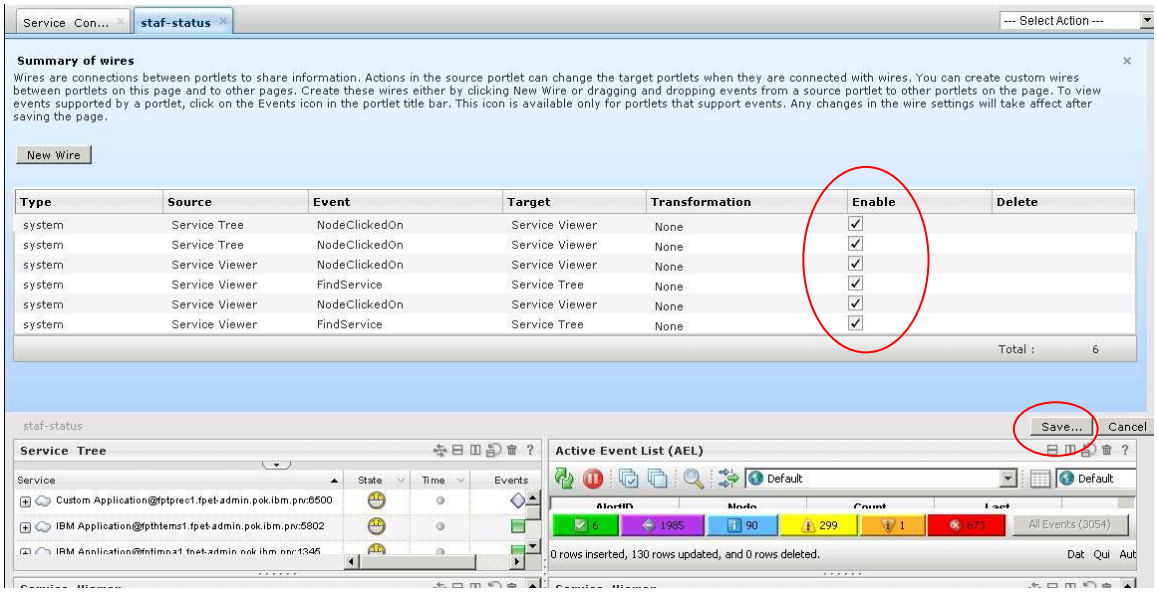


Figure 51 Remove “default” wires

8.3.17 Create one new nodeClickOn wire

The only wire needed for this example is one from the upper left Service to the upper right AEL. That wire is added by first editing the page. Then click on the wires icon highlighted below:

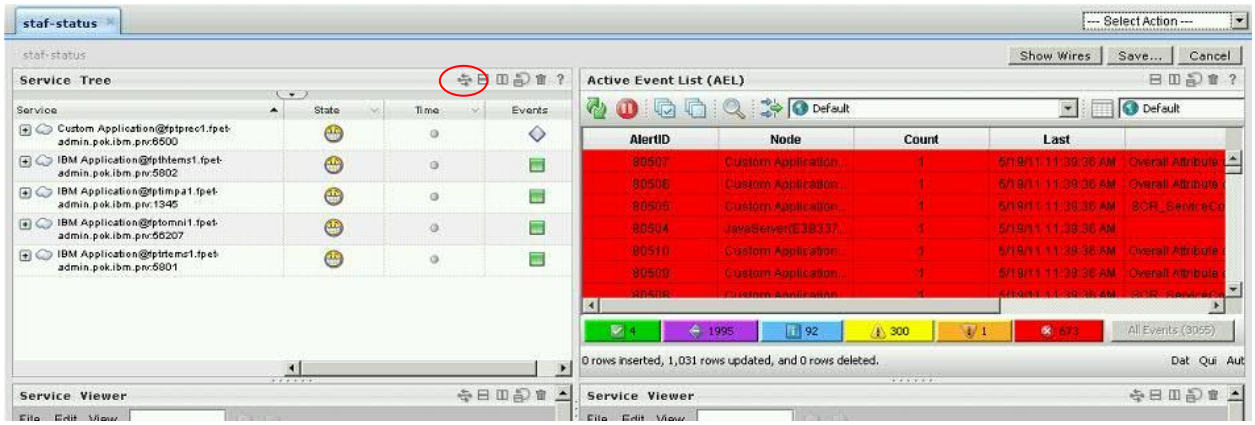


Figure 52 Add a new wire

From the service tree portlet click and drag the NodeClickedOn text hightlighted below and drag this to the AEL portlet on the right. Click “Ok” twice. This will create the desired wire automatically. Once complete, re-save the page. The purpose of this wire will be to display in context the events associated with the “STAF” elements in the service tree portlet.

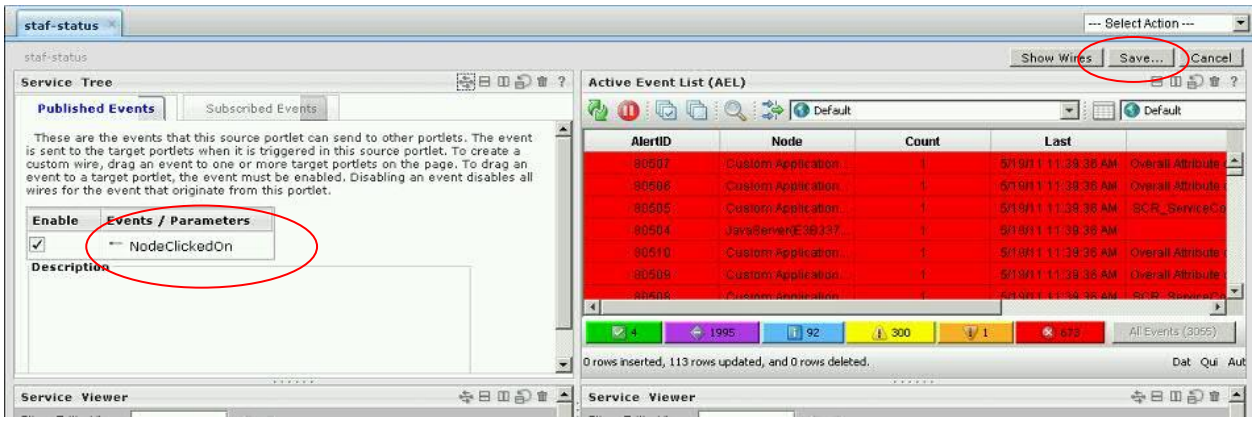

staf-status
--- Select Action ---
Show Wires
Save...
Cancel
Service Tree
Published Events
Subscribed Events
These are the events that this source portlet can send to other portlets. The event is sent to the target portlets when it is triggered in this source portlet. To create a custom wire, drag an event to one or more target portlets on the page. To drag an event to a target portlet, the event must be enabled. Disabling an event disables all wires for the event that originate from this portlet.
Enable
Events / Parameters
NodeClickedOn
Description
Active Event List (AEL)
Default
AlertID
Node
Count
Last
0 rows inserted, 113 rows updated, and 0 rows deleted.
Service Viewer
Service Viewer

Figure 53 Create a wire using drag and drop

8.3.18 The Dashboard (widget) portlet

The lower left portlet is a service viewer for the TestSub service and should look as follows:

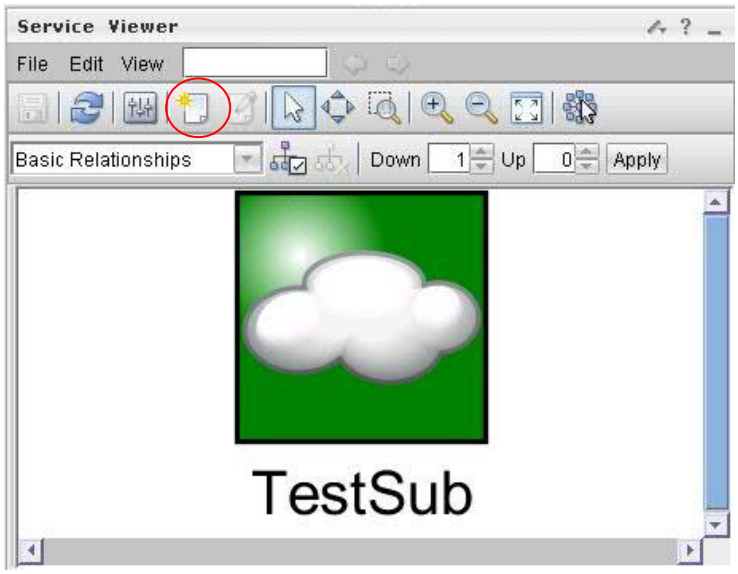


Figure 54 Current dasbord portlet view

Click New shown in Figure 54 Current dasbord portlet view.

“New” displays the view shown below:

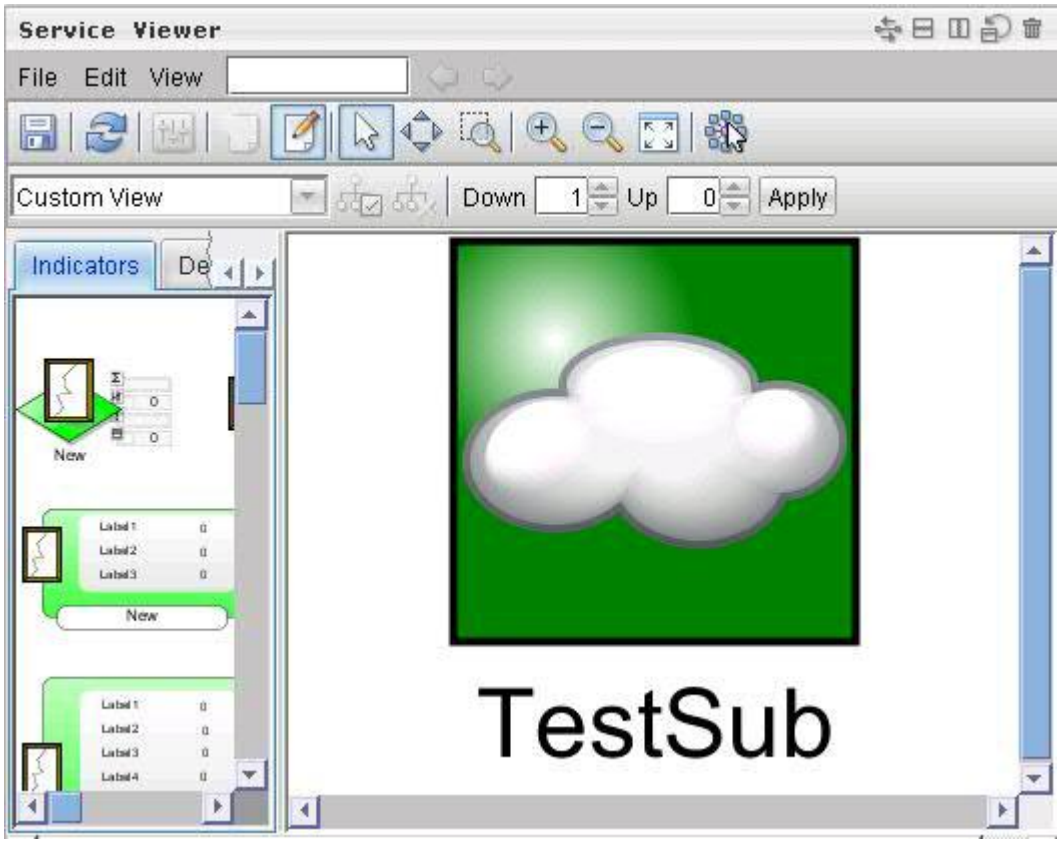


Figure 55 Defining a new service viewer using indicators

Click the desired indicator and click again in the portlet page to add it. Click on the view in the background which highlights it (shown in Figure 56 Add an indicator dashboard element). Once highlighted click delete on your keyboard to remove the orginal element(s) leaving only the new indicator.

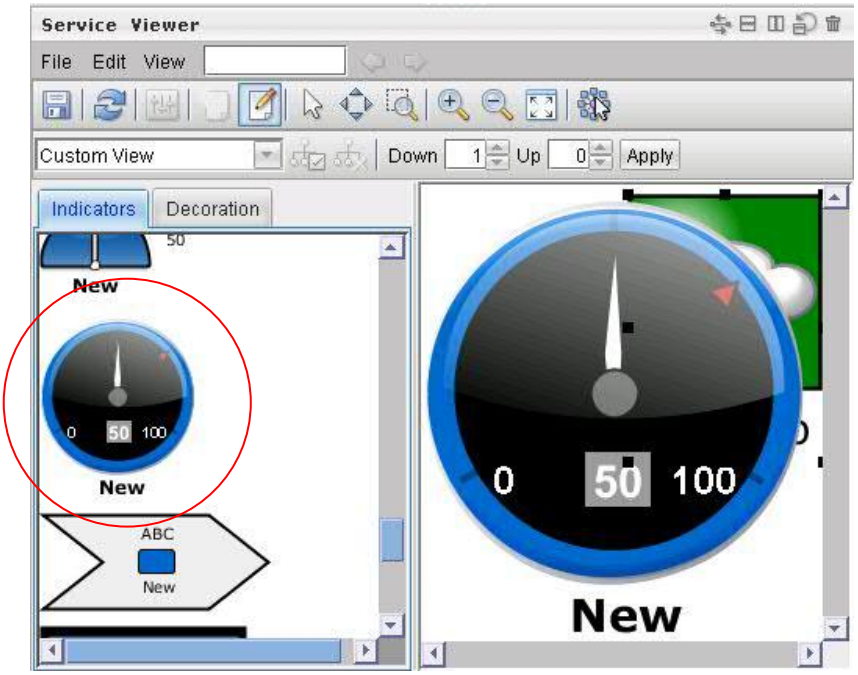


Figure 56 Add an indicator dashboard element

Adding a new indicator generates the display of the configuration panels for this widget. Select the "TestSub" service created earlier for this purpose and click Next.

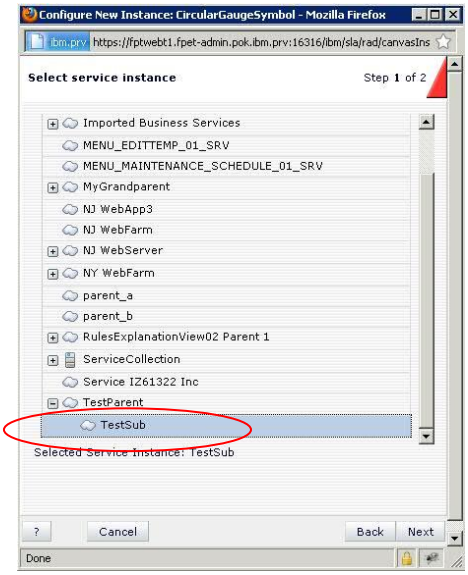


Figure 57 Indicator menu service instance

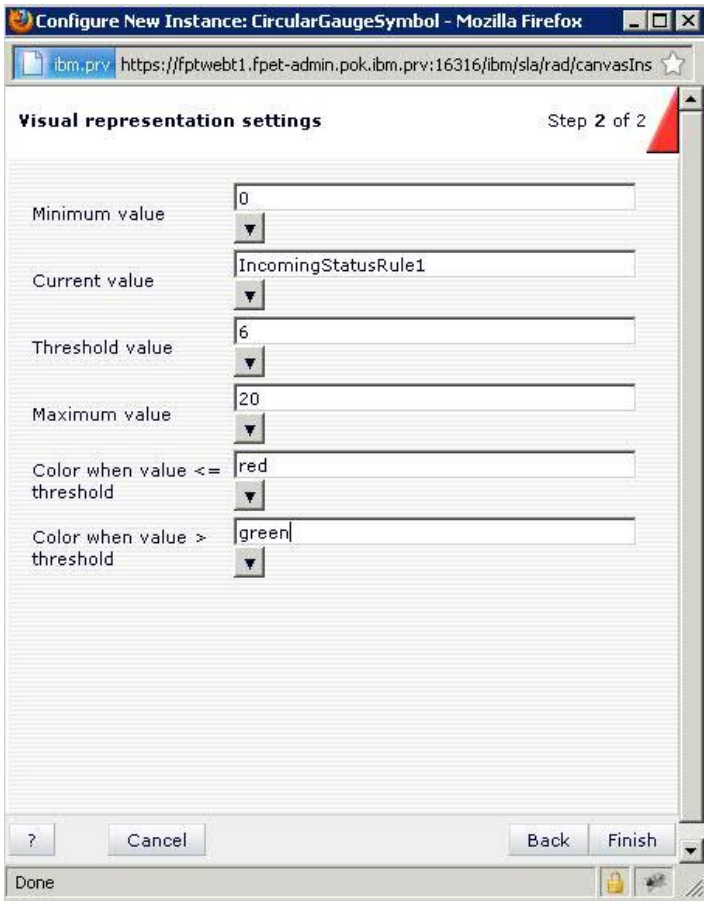


Figure 58 Enter the visual representation for this indicator

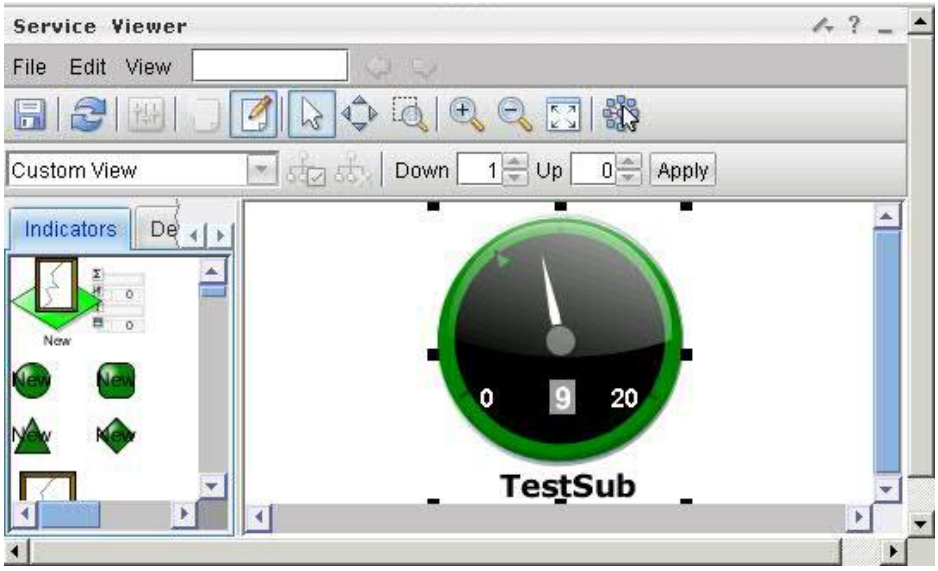
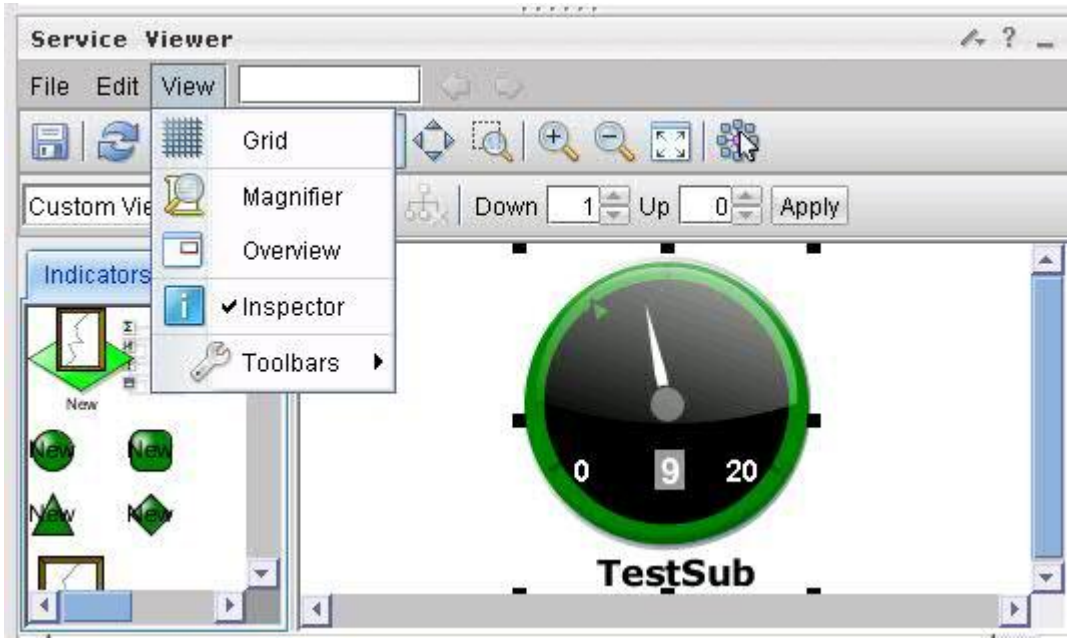


Figure 59 The indicator now contains count of STAF servers and the range (20)



If your image does not yet appear as shown above, and to set the name of the indicator it is necessary to edit the inspector panels. Click the following icon:

Click View, Inspector and if the inspector list is empty, click the indictor and bring the inspector back to the foreground.

On the inspector panel make the following changes:

1. Enter **StafService** in the ServiceInstanceName Value column

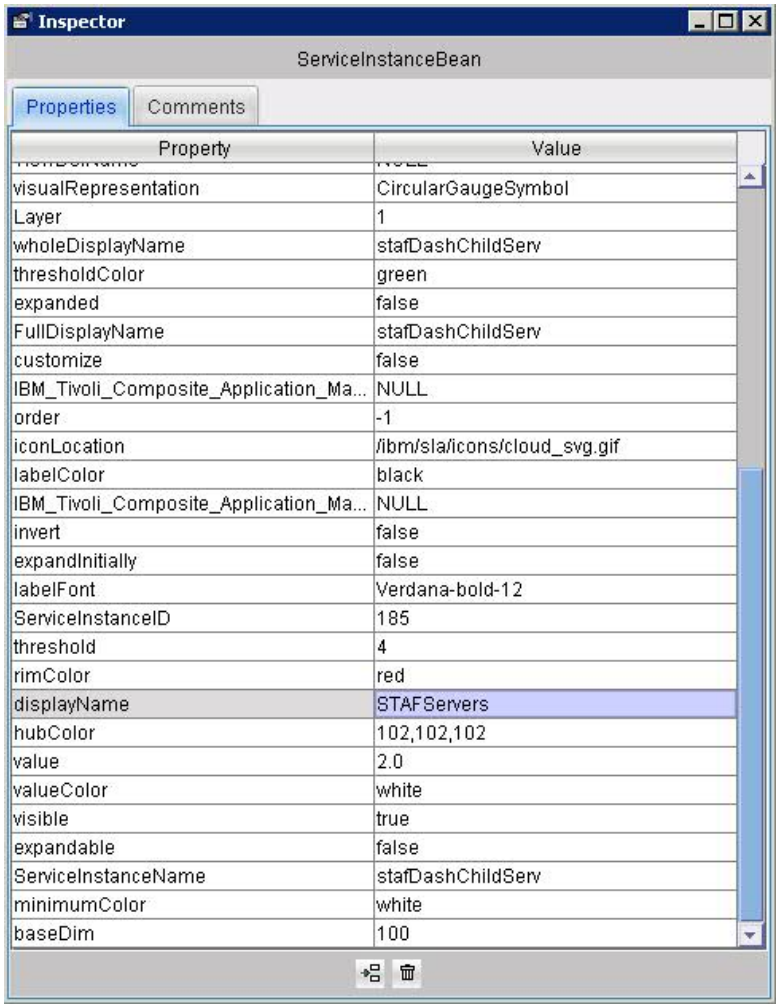


Figure 60 Set displayName to "StafServers"

The lower left portlet should now appear as shown below:

Figure 61 Complete widget dashboard

In the example shown above we altered the count to show when the status is below the threshold the dial appears red.

If you want to provide click actions for this widget. You can reference the following document for details:

https://www.ibm.com/developerworks/wikis/display/tivolibsm/Launching+Event+Displays+with+Dashboard+Widgets

8.3.19 Selecting your indicator/widget type

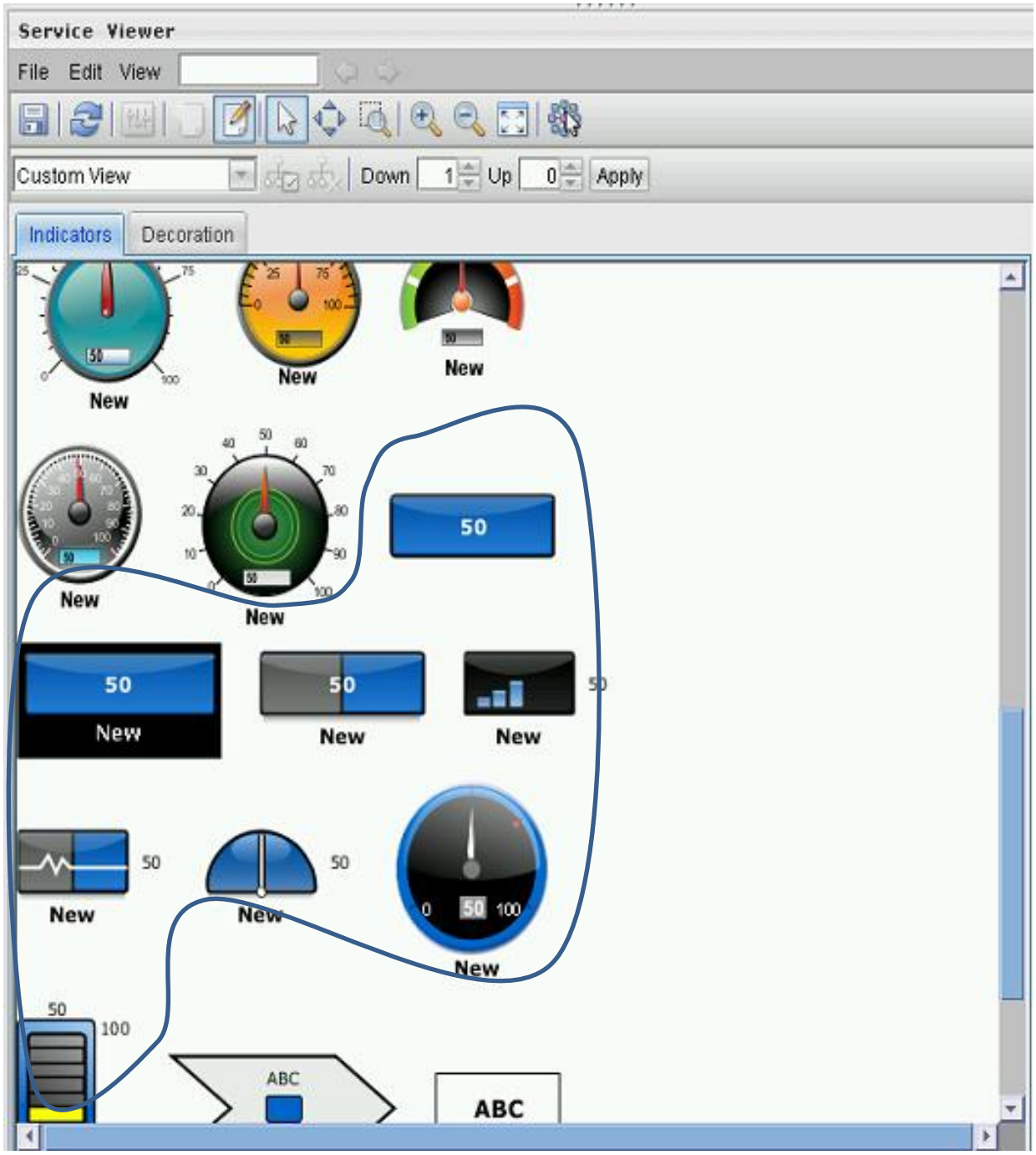


Figure 62 Widgets which support bi-directional negative status

The widgets above allow you to assign negative to positive status, or positive to negative status during the definition based on totals. In other terms if the totals are rising you may want to turn the status from green to red or red to green. Use these indicators for increased flexibility when defining your dashboards.

9 Changing the State GIF Images

Throughout this solution guide the service tree state GIF images have not been the defaults. As shown here:

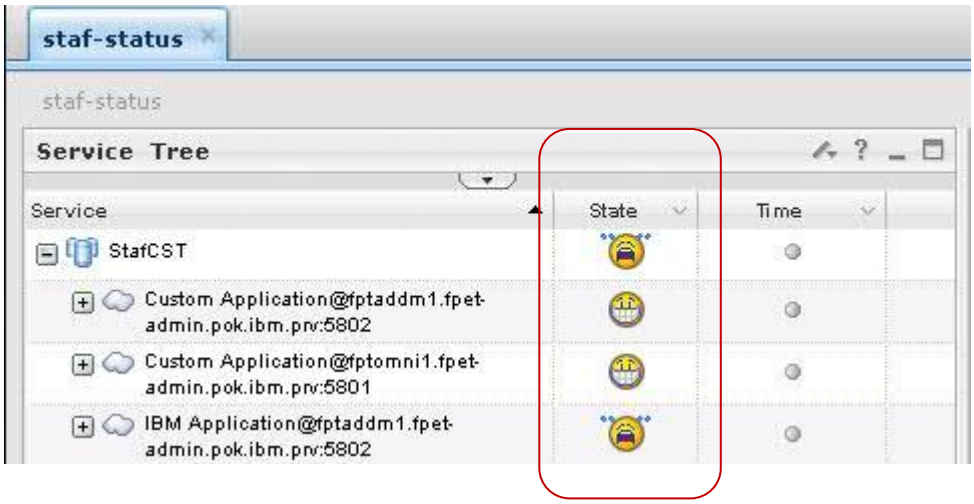


Figure 63 Service Tree State GIFs

If you would like to add your own “State” animations click on the Tree Template Editor icon shown below:

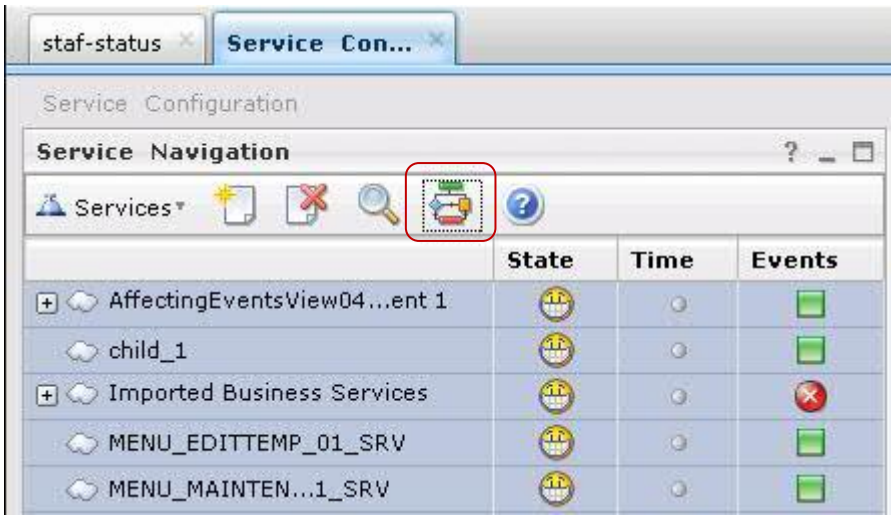


Figure 64 Tree Template Editor

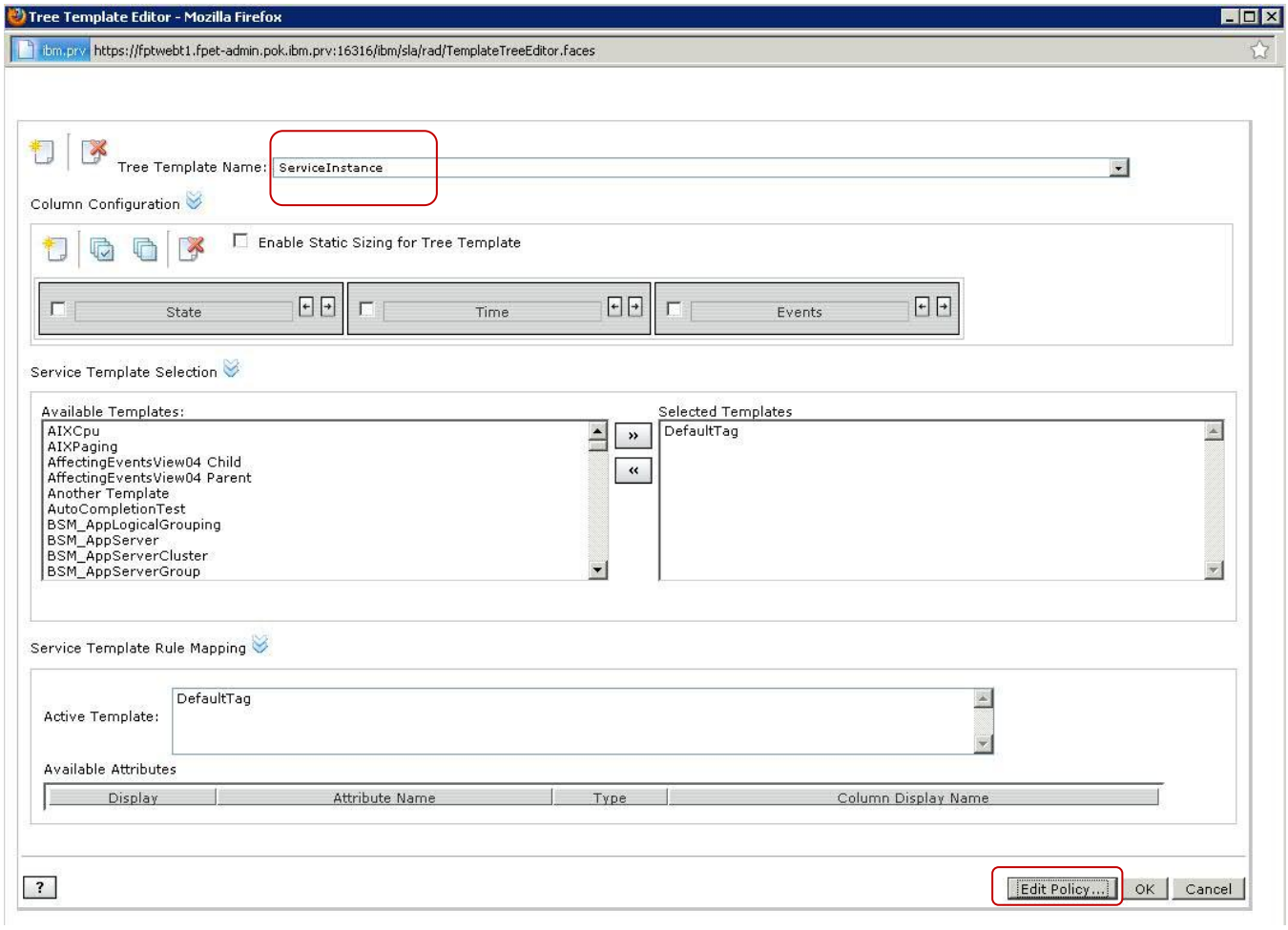


Figure 65 Select Template name and Edit Policy

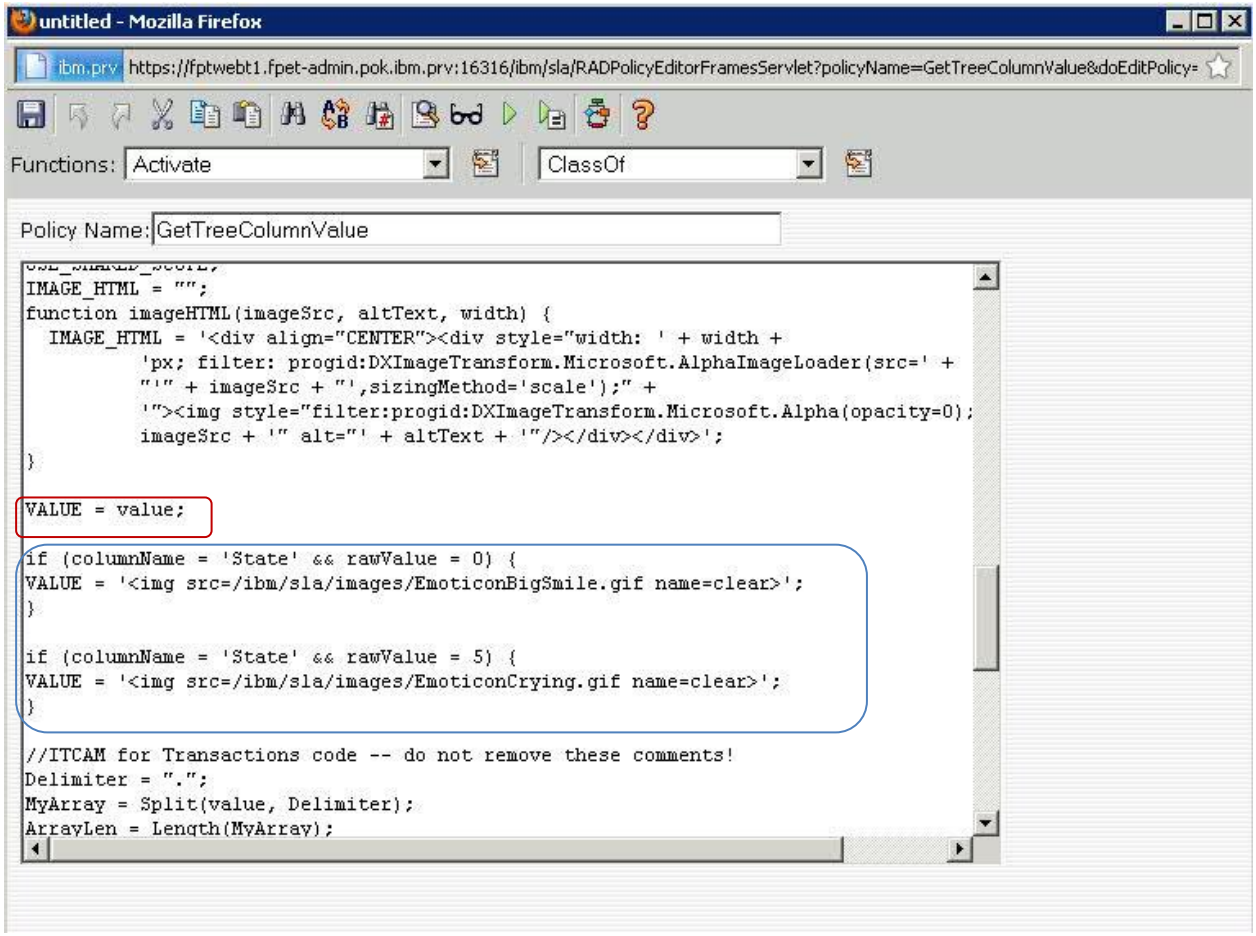


Figure 66 Edit Policy for gif's

As highlighted above the EmoticonBigSmile.gif and EmoticonCrying.gif are being used for events with rawValues of 0 and 5 only. You could provide unique gif's for rawValues 0..5. The two *if* statements highlighted in blue above must be added to change from the default gif's. These new statements must be placed under the VALUE = value; statement highlighted in red above. Any new gif's being added must be placed in the following directory:

%TBSM/TIPHOME%/tip/systemApps/isclite.ear/sla.war/images/

Once the images have been placed in that directory alter the tree template editor by adding one if-block for each rawValue image to be replaced.

When you have complete the adding of statements and gif's, click the save image .

10 Time Window Analyzer for STAF

TBSM Time Window Analyzer provides real time aggregate metrics for any KPI added to the TBSM metric history database.

An example of one such metric is shown in Figure 67 Time Window Analyzer.

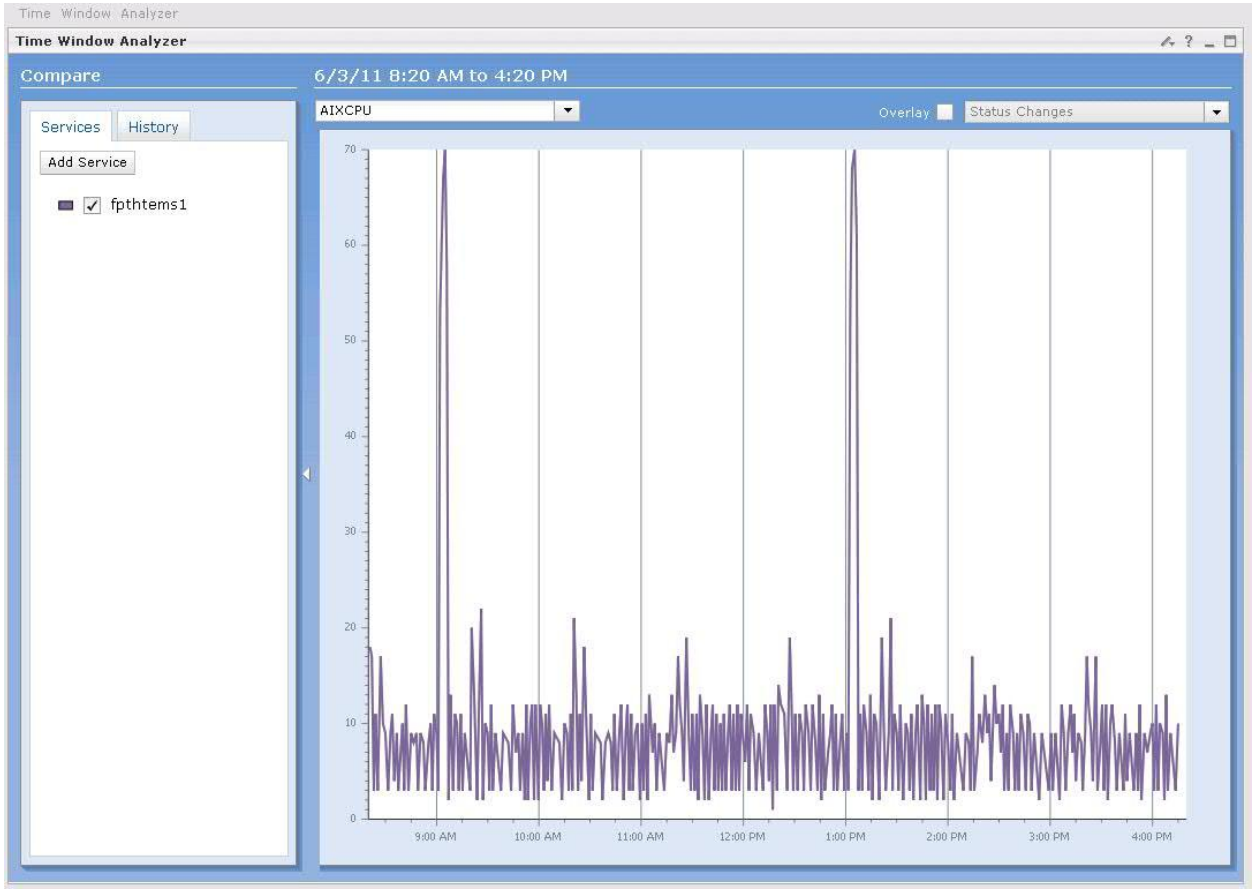


Figure 67 Time Window Analyzer

To add this support for the StafServers in this solution example, perform the following steps:

Display the template used to gather the metric.

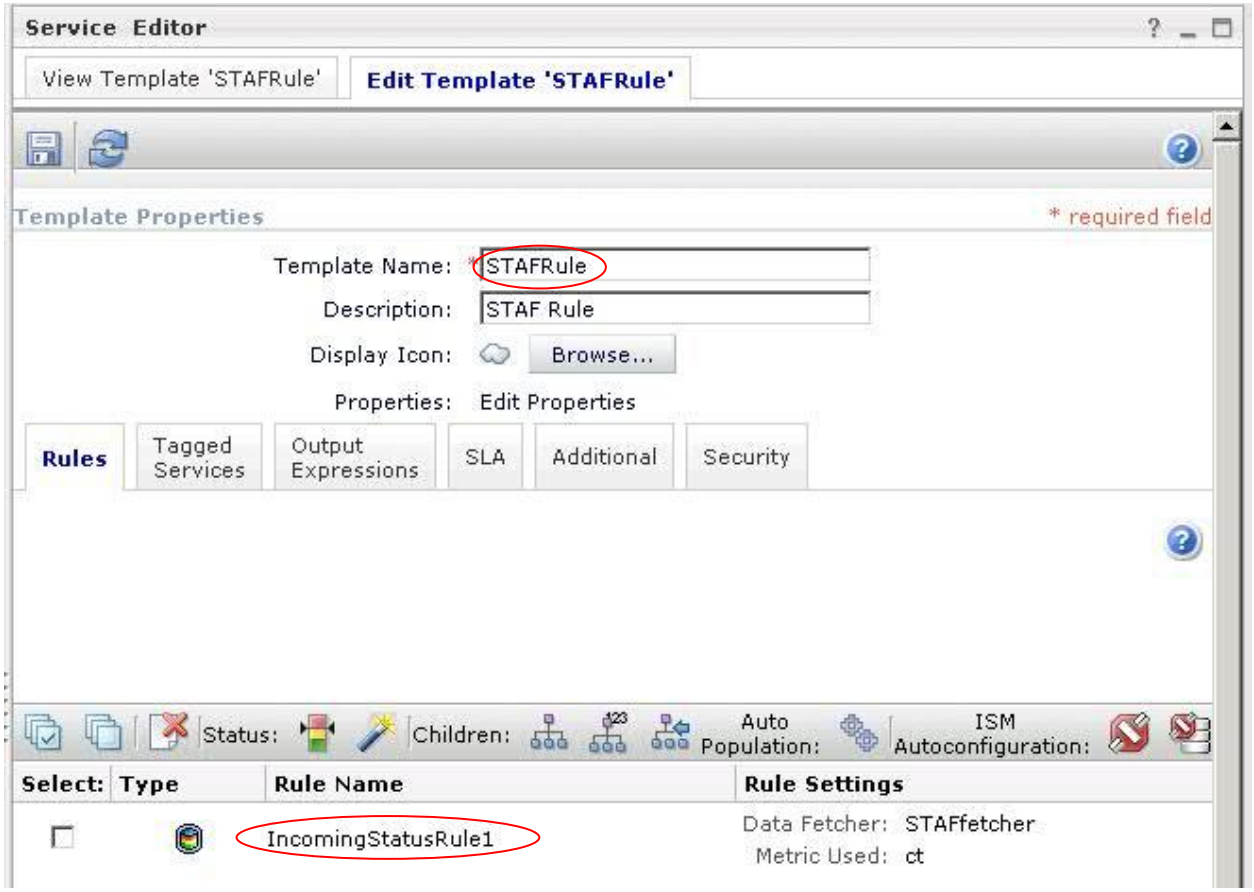


Figure 68 Record Template Name and Rule Name

From the TBSM data server change directory to %TBSM_HOME%/etc/.

Issue “scriptedAPIStartup.bsh” command

From radshell> prompt enter:
“addMetricMeta("STAFRule.IncomingStatusRule1","ct","Count",900,true,true);”

Data collection begins for the STAFRule, you need to wait a few minutes before displaying this metric data, this allows time to collect some of these data points.

Right-click on the dashboard object and click `Show>Compare` Service Metrics.

This displays the TWA panel for this metric, you could be monitoring your key KPI’s or the number of open tickets with this capability.

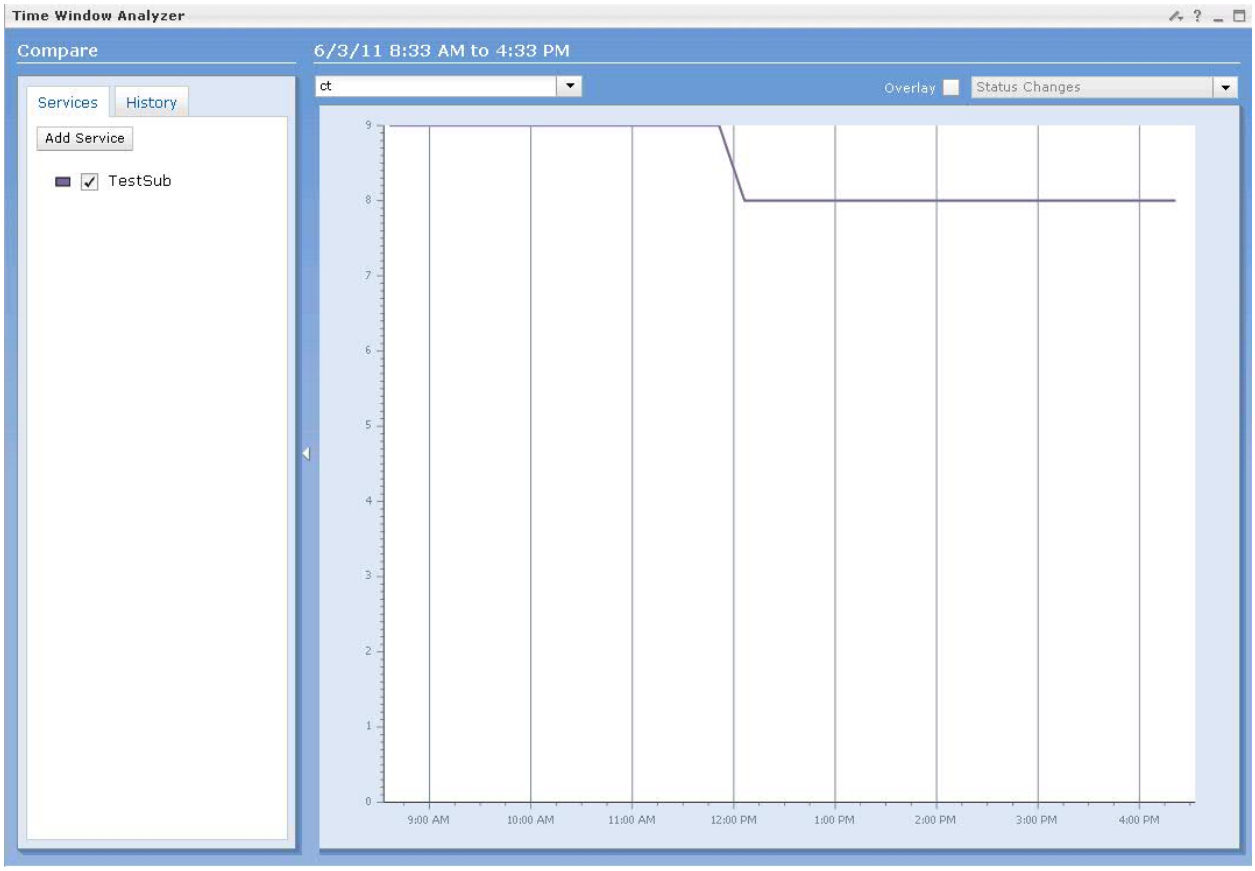


Figure 69 Time Window Analyzer for STAF services count (ct)

11 Tivoli Monitoring configuration (for STAF)

Integrating monitoring events between IBM Tivoli Monitoring and TBSM.

There are three steps in this process:
Run the Tivoli Monitoring DLA that is appropriate for your version of Tivoli Monitoring (tmsdla or KfwTmsDla).

The DLA enables Launch in Context to Tivoli Monitoring dashboards and enables event correlation in TBSM.

Load that DLA file into TADDM, TBSM, or both products.

Configure the Hub Tivoli Enterprise Monitoring Server (HTEMS) to forward events to the same probe that is used by TBSM/TADDM in this case.

Situation events from Tivoli Monitoring are automatically associated with objects that are being monitored by Tivoli Monitoring. TADDM also contains a monitoring coverage report that lists the Tivoli Monitoring managed nodes.

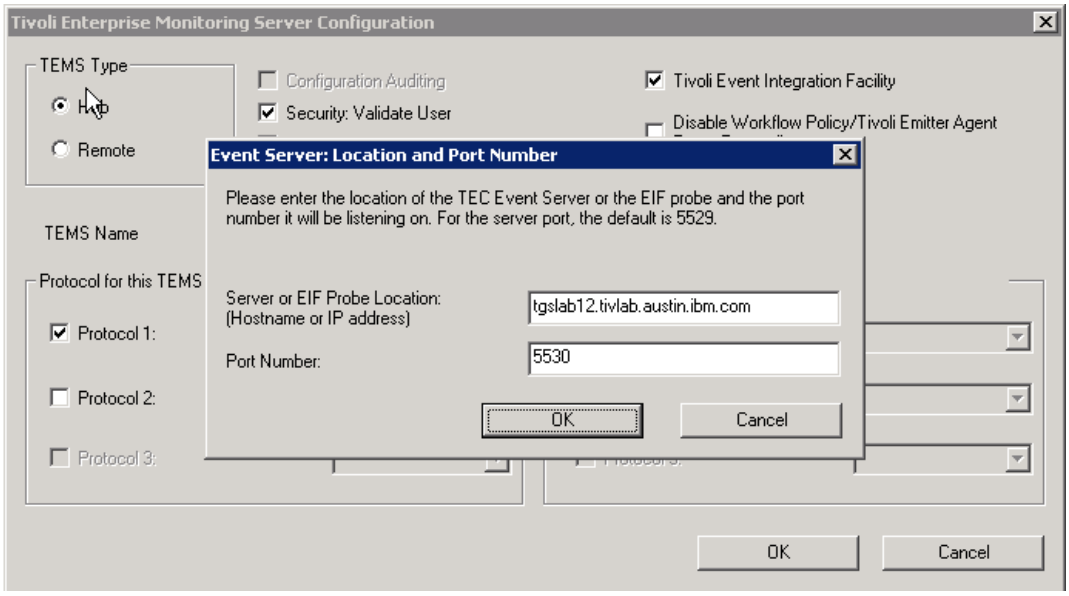


Figure 70 Configuring Tivoli Monitoring to send Events

11.1 Create a situation event for STAF

The following steps are performed on the Tivoli Enterprise Portal Server (TEPS). Right click on your Enterprise in the navigator and then click Situations.

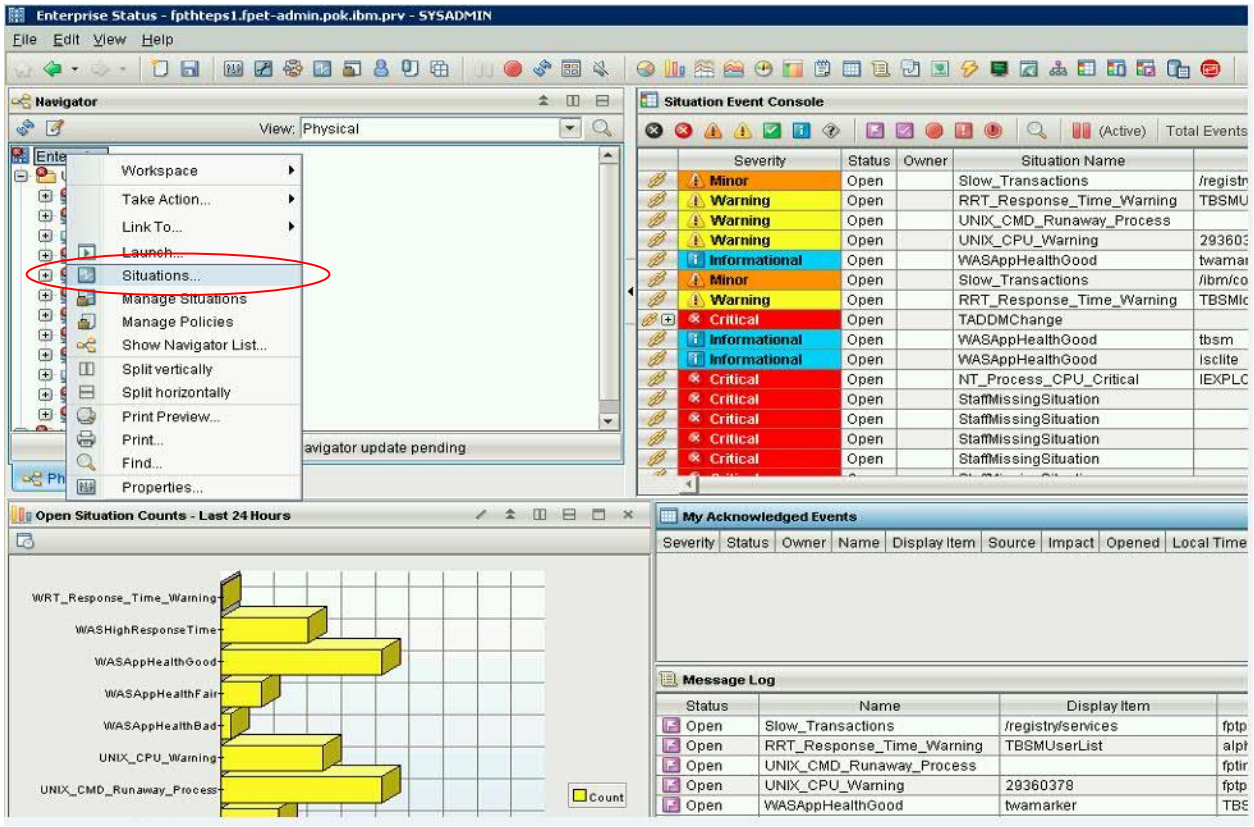


Figure 71 From the enterprise portal add a situation

11.2 Create a new Situation

Click on new situation icon, and enter what is shown in the following images.

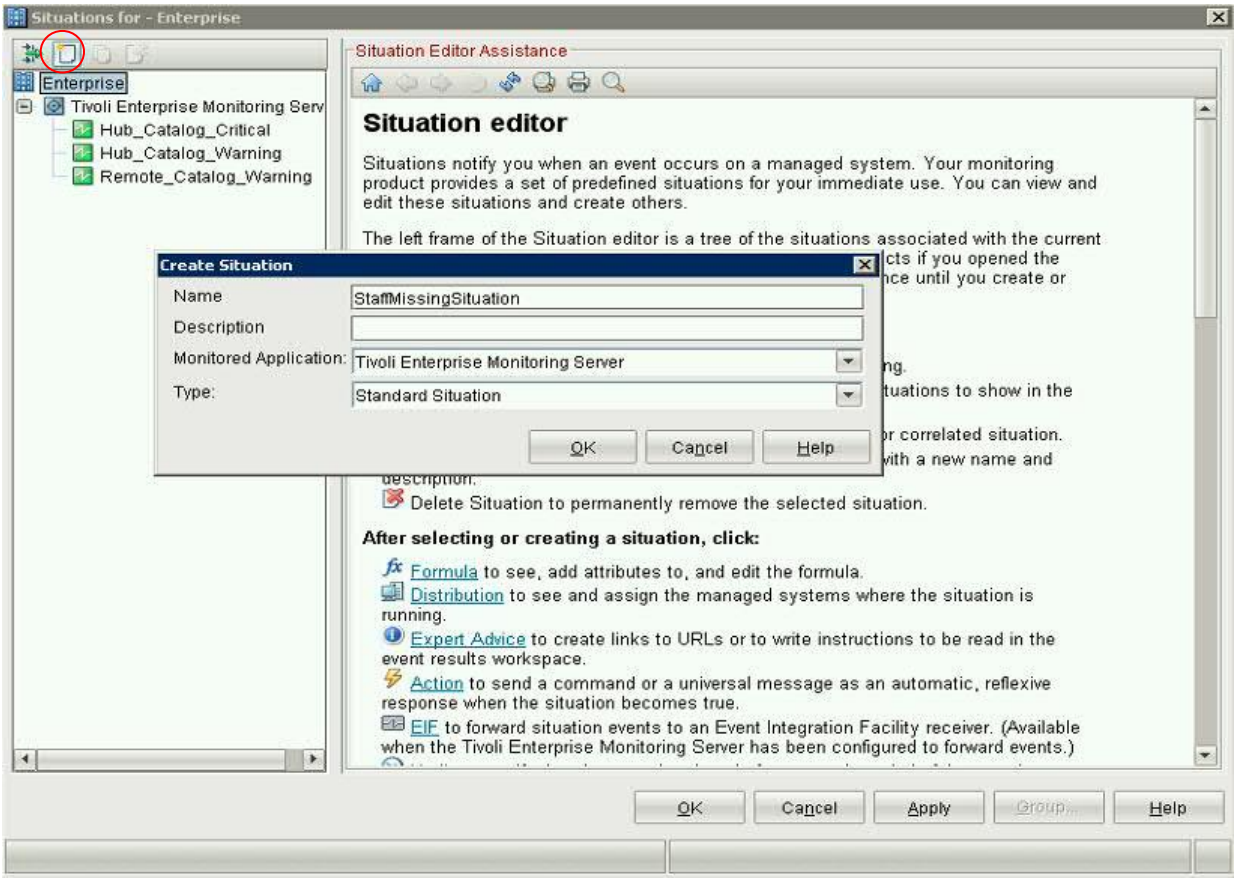

Situations for - Enterprise
Enterprise
Tivoli Enterprise Monitoring Serv
Hub_Catalog_Critical
Hub_Catalog_Warning
Remote_Catalog_Warning
Situation Editor Assistance
Situation editor
Situations notify you when an event occurs on a managed system. Your monitoring product provides a set of predefined situations for your immediate use. You can view and edit these situations and create others.
The left frame of the Situation editor is a tree of the situations associated with the current
Create Situation
Name
StaffMissingSituation
Description
Monitored Application:
Tivoli Enterprise Monitoring Server
Type:
Standard Situation
OK
Cancel
Help
Delete Situation to permanently remove the selected situation.
After selecting or creating a situation, click:
Formula to see, add attributes to, and edit the formula.
Distribution to see and assign the managed systems where the situation is running.
Expert Advice to create links to URLs or to write instructions to be read in the event results workspace.
Action to send a command or a universal message as an automatic, reflexive response when the situation becomes true.
EIF to forward situation events to an Event Integration Facility receiver. (Available when the Tivoli Enterprise Monitoring Server has been configured to forward events.)
OK
Cancel
Apply
Group...
Help

Figure 72 Set the situation name to “StaffMissingSituation”

11.3 Add the process command to identity the STAF process

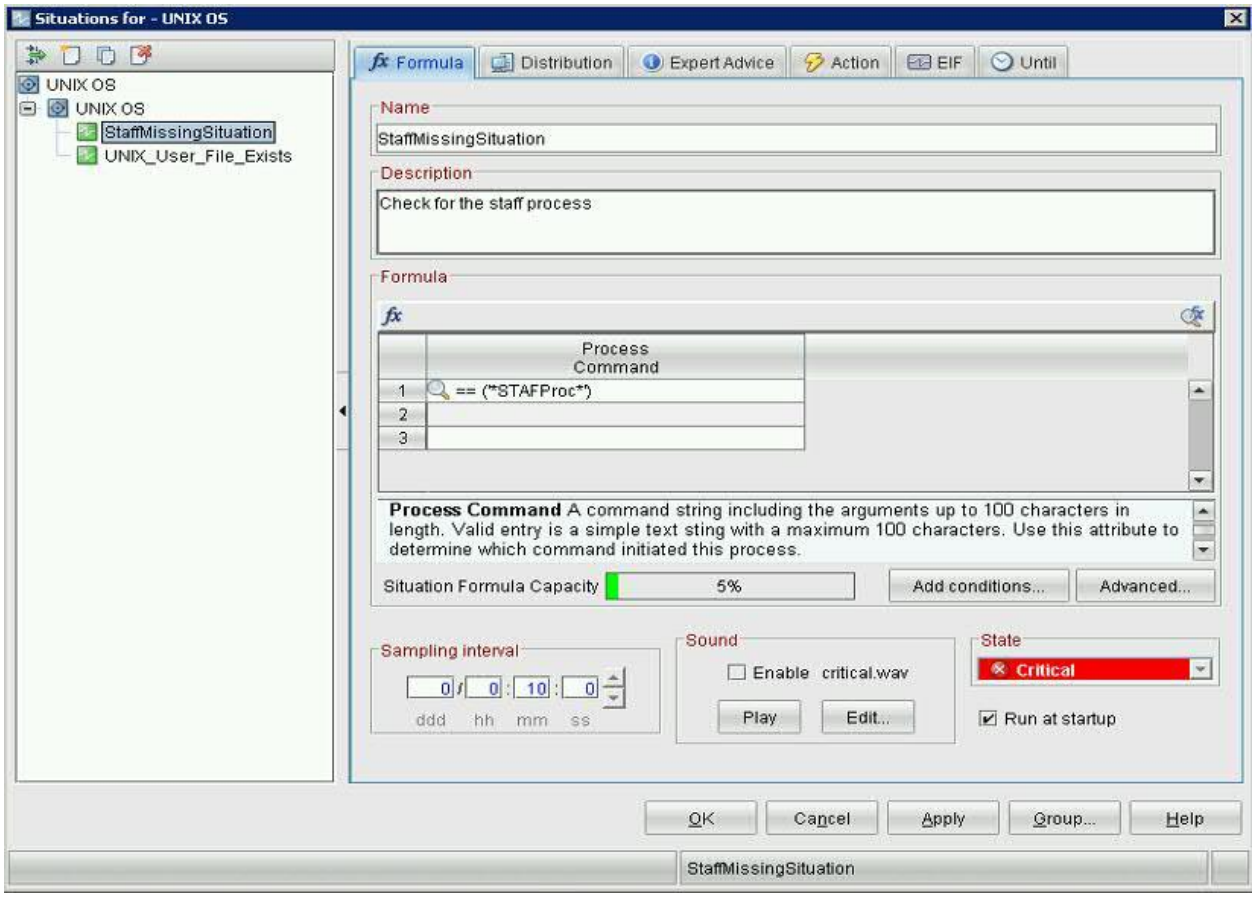


Figure 73 Set Process command as shown

The check for missing process command identifies STAFProc as the command name which results in the situations firing if missing. See *Figure 11 Define Custom Server* as a reminder how the STAFProc program name was used when creating the SystemTestAutoFrameworkSTAF custom server template. When entering the Process Command into the formula page enter only STAFProc, after entry it will then appear as ('STAFProc').

11.4 Assign the situation to the systems where STAF runs

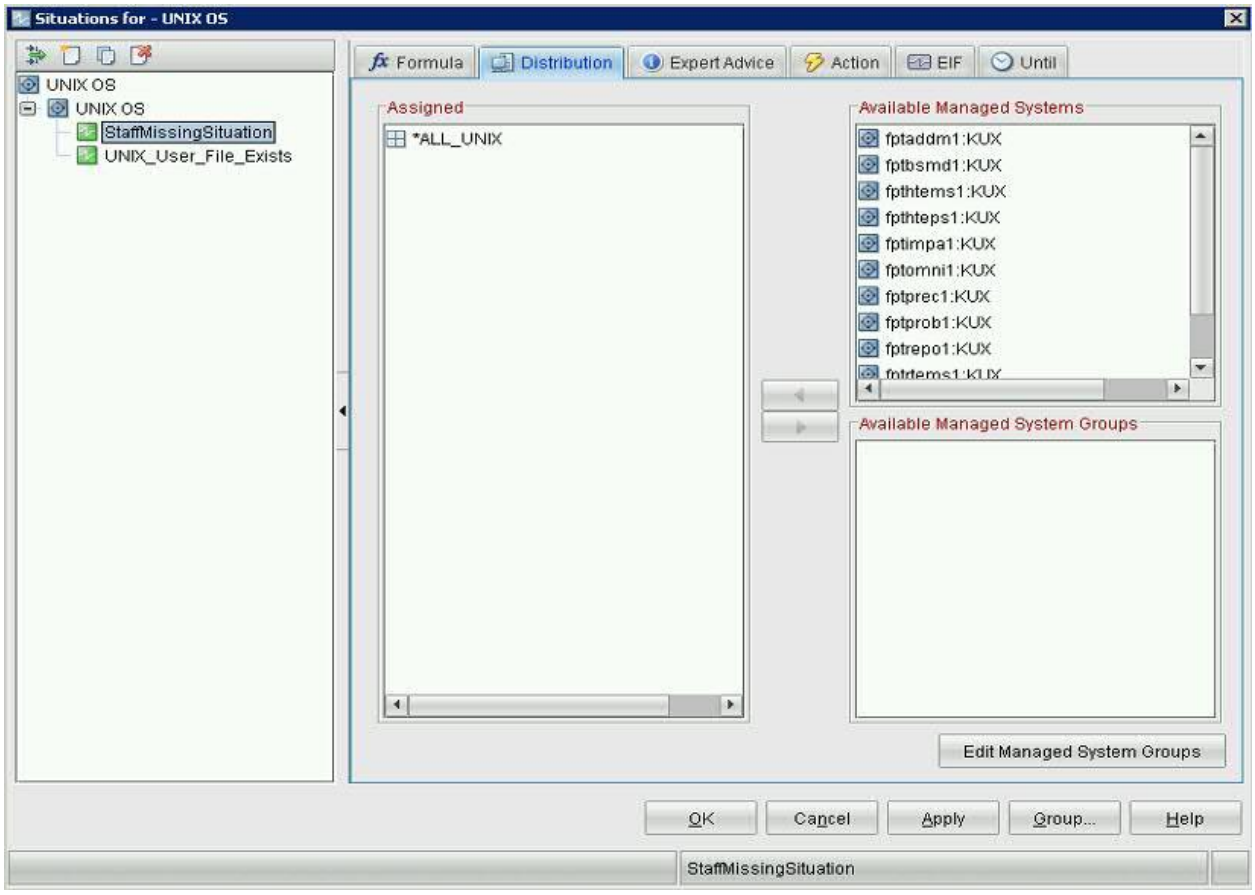


Figure 74 Assign to ALL_UNIX

For this solution STAF is assumed to be running only on UNIX and it is therefore assigned to all UNIX systems.

11.5 Set situation severity and target probe

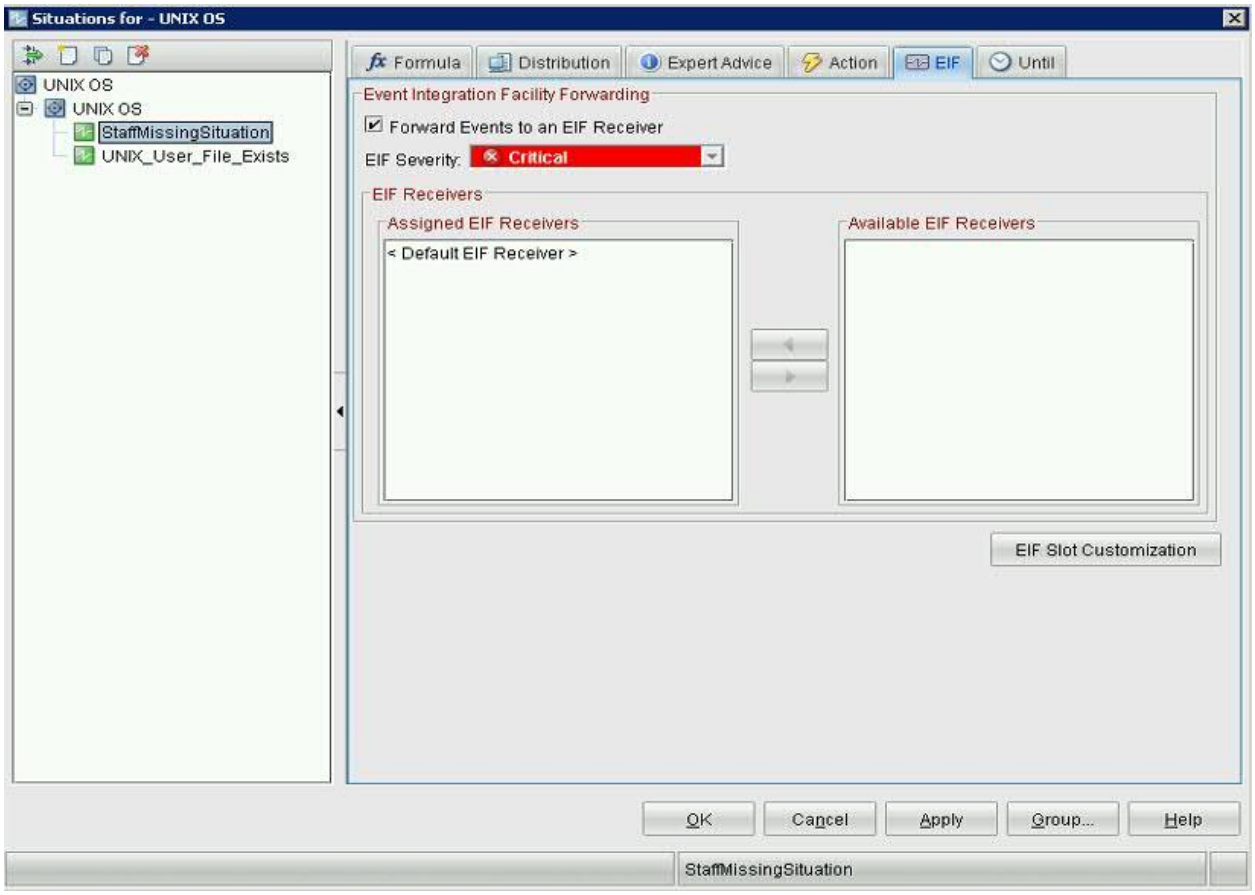


Figure 75 Set the EIF alert receiver

12 TADDM: Cleaning up dormant components

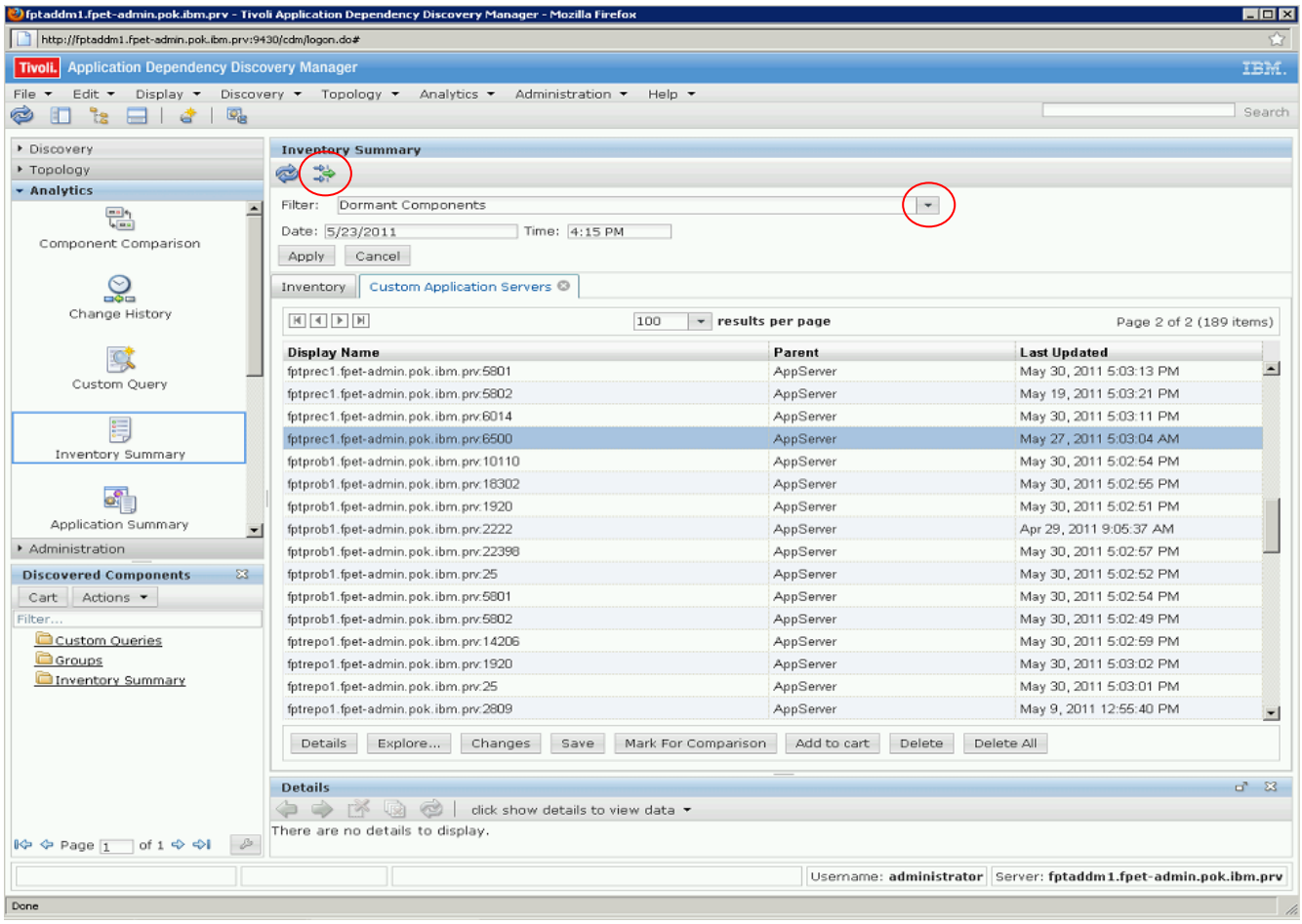

fptaddm1.fpet-admin.pok.ibm.prv - Tivoli Application Dependency Discovery Manager - Mozilla Firefox
http://fptaddm1.fpet-admin.pok.ibm.prv:9430/cdm/logon.do#
Tivoli. Application Dependency Discovery Manager
IBM.
File Edit Display Discovery Topology Analytics Administration Help
Search
Discovery
Topology
Analytics
Component Comparison
Change History
Custom Query
Inventory Summary
Application Summary
Administration
Discovered Components
Cart Actions
Filter...
Custom Queries
Groups
Inventory Summary
Page 1 of 1
Inventory Summary
Filter: Dormant Components
Date: 5/23/2011 Time: 4:15 PM
Apply Cancel
Inventory Custom Application Servers
100 results per page
Page 2 of 2 (189 items)
Display Name | Parent | Last Updated
fptprec1.fpet-admin.pok.ibm.prv:5801 | AppServer | May 30, 2011 5:03:13 PM
fptprec1.fpet-admin.pok.ibm.prv:5802 | AppServer | May 19, 2011 5:03:21 PM
fptprec1.fpet-admin.pok.ibm.prv:6014 | AppServer | May 30, 2011 5:03:11 PM
fptprec1.fpet-admin.pok.ibm.prv:6500 | AppServer | May 27, 2011 5:03:04 AM
fptprob1.fpet-admin.pok.ibm.prv:10110 | AppServer | May 30, 2011 5:02:54 PM
fptprob1.fpet-admin.pok.ibm.prv:18302 | AppServer | May 30, 2011 5:02:55 PM
fptprob1.fpet-admin.pok.ibm.prv:1920 | AppServer | May 30, 2011 5:02:51 PM
fptprob1.fpet-admin.pok.ibm.prv:2222 | AppServer | Apr 29, 2011 9:05:37 AM
fptprob1.fpet-admin.pok.ibm.prv:22398 | AppServer | May 30, 2011 5:02:57 PM
fptprob1.fpet-admin.pok.ibm.prv:25 | AppServer | May 30, 2011 5:02:52 PM
fptprob1.fpet-admin.pok.ibm.prv:5801 | AppServer | May 30, 2011 5:02:54 PM
fptprob1.fpet-admin.pok.ibm.prv:5802 | AppServer | May 30, 2011 5:02:49 PM
fptrepo1.fpet-admin.pok.ibm.prv:14206 | AppServer | May 30, 2011 5:02:59 PM
fptrepo1.fpet-admin.pok.ibm.prv:1920 | AppServer | May 30, 2011 5:03:02 PM
fptrepo1.fpet-admin.pok.ibm.prv:25 | AppServer | May 30, 2011 5:03:01 PM
fptrepo1.fpet-admin.pok.ibm.prv:2809 | AppServer | May 9, 2011 12:55:40 PM
Details Explore... Changes Save Mark For Comparison Add to cart Delete Delete All
Details
click show details to view data
There are no details to display.
Username: administrator Server: fptaddm1.fpet-admin.pok.ibm.prv
Done

Figure 76 Dormant Component Filter

> ⚠ **Be Aware**
>
> TADDM discovery does not delete a system or application if it does not respond. However, TADDM provides the Dormant Components filter option on the Inventory Analytics page for this purpose.

From the TADDM data management portal inventory summary, filter the list for dormant components. Highlighted above is a dormant STAF process. You can delete dormant components periodically to insure retired components are removed from your CMDB and TBSM service portlets.

13 Summary

Use TADDM to create and manage your business applications and services using the grouping composer. These groups are populated automatically by TADDM discover. TADDM provides an extensible discovery engine and the CMDB database. TBSM and its portlets provide a view of your business services, their health, and other key real time event/status information. TBSM and Tivoli Information Portal (TIP) also provide you with the custom dashboard functions that are required to maintain an overall view of the health of your key business services.

For more information see
http://www.ibm.com/developerworks/wikis/display/tivolibsm/Integrating+TBSM+4.2+with+Tivoli+Application+Dependency+Discovery+Manager

14 Appendix

14.1 Appendix A. Architecture

Computer	OS	Product name and version number
Tivoli Monitoring environment (hub monitoring server and portal server)	AIX 6.1	Tivoli Monitoring V6.2.2 Fix Pack 2
Tivoli Netcool/OMNIbus ObjectServer	AIX 6.1	OMNIbus V7.3
Tivoli EIF Probe	AIX 6.1	TBSM V4.2.1 Fix Pack 2
Tivoli Application Dependency Discovery Manager	AIX 6.1	TADDM V7.2.1 (GA)
Tivoli Netcool/Webtop	AIX 6.1	Tivoli Netcool/Webtop V2.2.x (TBSM V4.2.1 Fix Pack 2)

14.2 Appendix B. Information centers

Products	Information center
TBSM data server, TBSM discovery library toolkit, Tivoli EIF Probe, Tivoli Netcool/Webtop (TBSM dashboard)	IBM Tivoli Business Service Manager
IBM Tivoli Composite Application Manager Agents (PeopleSoft, DB2)	IBM Tivoli Composite Application Manager Agents
Tivoli Application Dependency Discovery Manager	IBM Tivoli Dependency Discovery Manager
Tivoli Netcool/OMNIbus ObjectServer	IBM Tivoli Netcool/OMNIbus Information Center
IBM Tivoli Monitoring	IBM Tivoli Monitoring

14.3 Appendix C. Acronyms

Acronym	Definition
AEL	Active Event List
CI	Configuration Item or an object in TADDM/TBSM
crviewer	DLT Component Registry Viewer
CST	Custom Server or Custom Server Template
DLT	Discovery Library Toolkit
EIF	Event Integration Facility
IDML	International Development Markup Language
TADDM	Tivoli Application Dependency Discovery Manager
TBSM	Tivoli Business Service Manager
DLT	Discovery Library Toolkit
TWA	Time Window Analyzer

15 Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions

of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM‘s application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM Corp. 2012. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2012 All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

15.1 Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.
Other company, product, or service names may be trademarks or service marks of others.